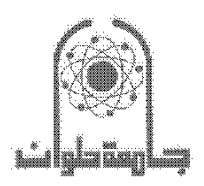


Faculty of Engineering
Mechanical Department – Mechatronics Division

IDENTIFICATION, ESTIMATION AND PIP CONTROL APPLIED TO MECHANICAL AND HEAT TRANSFER SYSATEM

Ahmed Rafat Radwan
Gamal Ahmed Mostafa
Ahmed Negmeldin Salama
Hassan Ahmed Hassan
Amr Eid Shehata
Peter Fawzy Salama

Supervised by

Dr. Essam-Eldin Mohammed Shaban

July 2008

ACKNOWLEDGEMENT

First and foremost, I feel always indebted to God, the most gracious and the most merciful.

I would like to express my deepest thanks and gratitude to my supervisor **Dr. Essam Shaban**, *Assistant Professor in Faculty of Engineering, Mattaria, Helwan University*, for his valuable guidance, support and encouragement throughout this research work. I am also grateful to **Eng. Mohammed Elsayed**, *Demonstrator in Faculty of Engineering, Helwan, Helwan University*, for his great guidance hardware integration in both projects. Finally, a special thanks are due to *Prof. Dr. Abdel-Haleem Bassiouni, Head of Mechatronics Division*, and *Prof. Dr. Mokhtar Bakr, Vice Dean, Helwan University*.

Special thanks to the Cellopack Company their support and offering their machines for as the first demonstration.

We would like to forward our deepest appreciation to our families for their painstaking and encouragement given over such a long period of time.

God bless every hand that helped me to accomplish this work.

Table of Contents

List of Figures

CHAPTER 1

INTRODUCTION

Automation plays an important role in our daily life. It is merged in most of application we use every day such as traffic lights, automatic doors, elevators, cars and many other applications. It is also extensively used in industry in order to improve quality and/or increase quality. Also, automation is essential for safety and security in companies such as fire alarms and CCTV cameras.

One possible definition for automation is: *"the use of control systems such as computers or logical drives to control machinery and/or any other processes to reduce the need for human intervention"*. In the scope of industry, automation is a step beyond mechanization, whereas mechanization provided human operators with machinery to assist them with the physical requirements of work; however automation greatly reduces the need for human sensory and mental requirements as well. Nowadays, industrial automation is becoming increasingly important in the manufacturing process because computerized or automated machines are capable of handling repetitive tasks quickly

and efficiently. Moreover, they are capable of completing mundane tasks that are not desirable to workers.

Several advantages can be achieved by automation, they are:

(1) increasing productivity,

(2) reducing labor cost,

(3) mitigate the effects of labor shortages,

(4) almost eliminating routine manual and clerical tasks,

(5) improving workers safety,

(6) improving productivity,

(7) reducing manufacturing lead time,

(8) accomplishing processes that cannot be done manually,

(9) reducing unit cost.

Since automation and mechatronics are related to each other, it is important to define the meaning of mechatronics as: *"The synergistic integration of mechanical engineering, with electronics and intelligent computer control in the design and manufacturing of industrial products and processes"*. The study of mechatronic systems can be divided into the following areas of specialties:

1. Physical Systems Modeling.
2. Sensors and Actuators.
3. Signals and Systems.
4. Computers and Logic Systems.
5. Software and Data Acquisition.

The present work shade the light on the True Digital Control (TDC) approach to control system for which the controller design is carried out entirely in discrete time, starting

from the identification and estimation of suitable linearised model to the real-time implementation of the final control law. Therefore, as a first step, it is required to find a linearised representation of the dynamic system based on a transfer function (TF) model, with parameters identified and estimated from measured data.

Classical approaches to control systems design consider only continuous time methods. However, with the advent of the digital computer, such continuous time techniques have been modified for use in sampled data systems. Here, the classically derived control law is converted into digital form (e.g. using finite difference approximations) and implemented as shown in Figure (1.1).

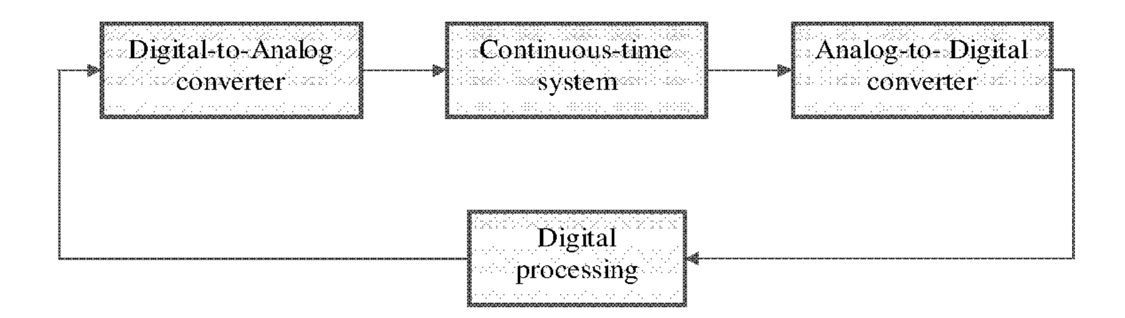


Figure 1.1 Sampled data control system

Alternatively, in view of the wide availability of digital computers, it is convenient to carry out the entire control system design procedure, from model identification through to the implementation of the final algorithm, in discrete time. This approach is sometimes called *direct digital control or true digital control* TDC.

Experimental or data based modeling consists of the development of mathematical models of dynamic systems on the basis of measured data from laboratory/field experiments. The data are utilized to both *identify* the model structure (i.e. the orders of

the polynomials in a transfer function model and the size of the pure time delay) and to *estimate* the parameters which characterize the dynamic behavior of the system.

Sophisticated *time-series* methods are required for this. So, neither the underlying physics nor internal life of the system need necessarily play a role in model generation. In contrast to physical modeling there are procedures for experimental modeling in which the modeling can be wholly or partially automated.

The use of automated systems has been extended in recent years to include not only usual industries but also *packaging industries* and *tracking systems* which may have series of repetitive operations. This automated process provides several benefits such as improving the quality, increasing the productivity and saving human life when working in such hazardous environment.

In spite of sophisticated capabilities of such automated systems, they need well designed controller that can comprise several surrounding changes. Therefore, the significance of true digital control TDC has arisen, in which the design of control system is accomplished explicitly in discrete time. TDC approach is based upon estimating the transfer function that best model the system ``*data-based modeling*``, and the design of the control algorithms that keep such automated systems under control.

So most of mechatronic systems are working automatically and these systems are called automated systems. Also, automation is the technology dealing with the mechatronics and computers for production of goods and services. The next figure, Figure (1.2), shows an example of automated system which can be used in vehicles, for which the car speed can be adapted automatically without any human intervention by means of feedback controller.

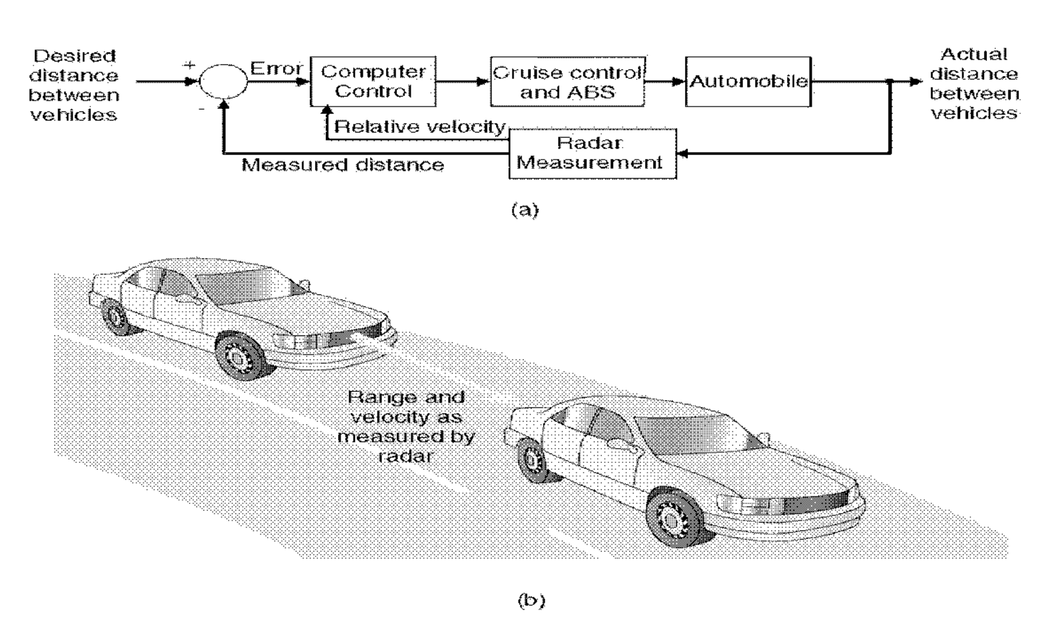


Figure 1.2 Using radar to measure distance and velocity to autonomously maintain desired distance between vehicles. (Adapted from *Modern Control Systems,* 9th ed., R. C. Dorf and R. H. Bishop, Prentice-Hall, 2001)

1.1 Controller in Industry

The key of automation is the control system, which can be defined as *the device that provides the system with command input to follow desired output.* The main objectives of controllers are:

(1) following trajectory.

(2) improving performance of controlled system.

The traditional types of controllers were analog; however many of today's controllers can handle digital signals.

The most common types of controllers used in industry are *Programmable Logic Controller* PLC and *Proporional-Integral-Derivative* PID. The first type, PLC, is a

digital computer used for automation of industrial processes, such as control of machinery on factory assembly lines

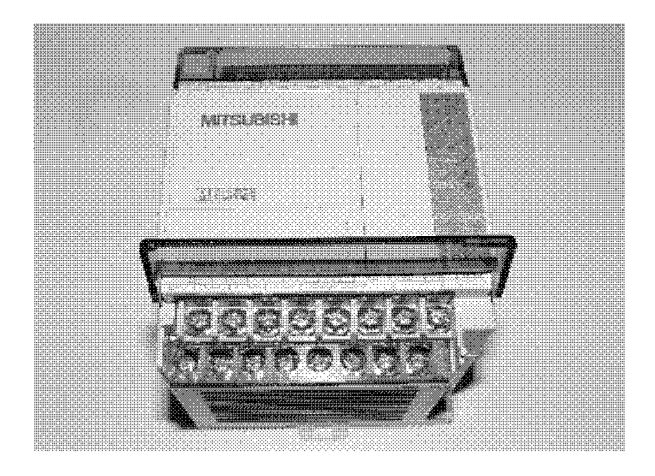


Figure 1.3 A snapshot of MITSUBISHI PLC unit

The main Advantages of control panel that is based on a PLC controller compared to a conventional process control system can be summarized in the following points:

(1) Number of wires needed for connections is reduced by 80%.

(2) Consumption is greatly reduced because PLC consumes less than a bunch of relays.

(3) Diagnostic functions of a PLC controller allow for fast and easy error detection.

(4) Change in operating sequence or application of a PLC controller to a different operating process can easily be accomplished by replacing a program through a console or using a PC software (not requiring changes in wiring, unless addition of some input or output device is required).

(5) Reliability of a PLC is greater than that of an electro-mechanical relay or a timer.

However, disadvantages of PLC are:

(1) Relative to electro-mechanical relays, PLC's are relatively fragile instruments easily damaged by water leaking on them.

(2) PLC's are less robust than a good hardwired system.

(3) Technological level of competence is much higher for installation and maintenance.

Proportional-Integral-Derivative (PID) controller is a generic control loop feedback mechanism widely used in industrial control systems. PID controller attempts to correct the error between a measured process variable and a desired setpoint by calculating and then outputting a corrective action that can adjust the process accordingly.

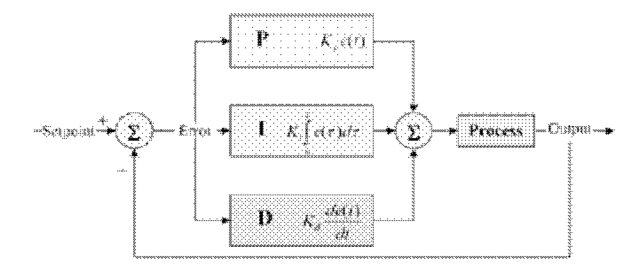


Figure 1.4 PID control loop

The proportional part is responsible for reducing the large part of the overall error. The integral part is responsible for reducing the final error by summing even a small error over time to produce a drive signal large enough to move the system toward a smaller error. However, derivative part is responsible for counteracting both proportional and integral parts when the output changes quickly. Derivative action help reducing the overshoot and it has no effect on th efinal error.

Although PID controllers have the ability to control dynamic system online efficiently, they have the following disadvantages:

(1) Its performance reduces considerably when the system has time delay and/or has more than unity order

(2) It consumes time since the gains are determined by trial and error.

(3) It needs training since it depends on well-trained and experienced engineers.

1.2 PIP Controller

The above types of controllers (PLC and PID) can overcome input and/or output disturbances within a very limited range. Therefore, a new methodology is introduced in this thesis in order to efficiently overcome disturbances and/or deal with systems having higher order and/or higher time delay. This new control algorithm namely Proportional-Integral-Plus for which the control law is based on an estimated discrete-time model showed good-enough performance for applied system since 1998. This methodology has been first published by Young *et al.*, 1987.

The PIP controller can be interpreted as a logical extension of PI/PID controllers but with inherent model-based predictive control action, for which additional dynamic feedback and input compensators are introduced automatically when the dynamic system has second order or higher, see Figure (1.4).

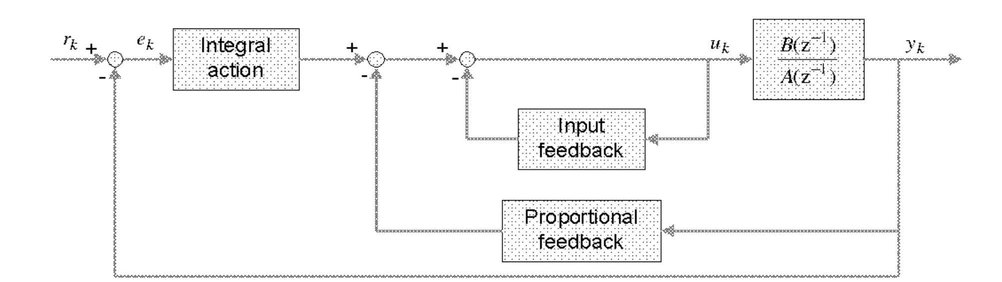


Figure 1.5 PIP control in block diagram feedback form

Figure (1.2) illustrates the simplest case of a single input, single output system, represented by a discrete-time transfer function model,

$$y_k = \frac{B(z^{-1})}{A(z^{-1})} u_k \tag{1.1}$$

where y_k is the controlled output variable, u_k the control input and,

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}$$
$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m} \quad (1.2)$$

in which $a_1 \cdots a_n$ and $b_1 \cdots b_m$ are parameters, while z^{-1} is the backward shift operator, i.e. $z^{-i} y_k = y_{k-i}$. In the present research, equations (1.1) and (1.2) are identified and estimated from experimental data using *fminsearch* function existed in Matlab®.

Here, an appropriate structure for the transfer function needs to be defined, i.e. the order on the numerator *n*, the number of numerator terms *m* and the time delay triad δ. The main statistical measures employed to help determine these values are the coefficient of determination R_T^2, based on the response error, which is a simple measure of model fit.

Although similar to conventional PI/PID design in structural terms, model-based PIP control has numerous advantages: in particular, it exploits the power of state variable feedback methods, where the vagaries of manual tuning are replaced by pole assignment or Linear Quadratic (LQ) design. Only, the pole assignment will be used in this thesis. Over the last few years, such linear PIP control systems have been successfully employed in a wide range of applications. An incomplete list includes: Chotai *et al.* (1991); Young *et al.* (1994); Dixon and Lees (1994); Taylor *et al.* (1998b, 2000b, 2000c, 2004a, 2004b); Gu *et al.*, (2004); Shaban *et al.*, 2007.

The proportional-integral-plus (PIP) controller has a great flexibility in design terms. For example, it can be applied to multivariable systems described by backward shift operator models. The present thesis emphasises the flexibility of the PIP approach in the context of the control structure selected for implementation. It has been demonstrated that the

choice of structure has important consequences for the robustness of the final design to parametric uncertainty and its disturbance rejection characteristics.

The thesis discusses the performance and application of four types of controllers; they are the premitive P- and I-controllers and the more advanced PI- and PIP controllers applied for two types of demonstrators. The first demonstrator is Hot-Air Temperature control of a Natural Gas Burner in Cellopack company located in 6^{th} of October city. The second demonstrator is a prototype of one-degree of freedom tracking mechanism.

As will be shown in the proper chapter P-, I- and PI controllers have their own limitation, however PIP shows great flexibility in design and good performance in application in both demonstrators.

1.3 Demonstrators

The thesis identifies models and apply P-, I-, PI- and PIP control systems for two application examples, they are:

- **Hot Air Control (Heat Transfer System):** This is the first example which based on controlling the temperature of the hot air supplied from a natural gas burner. The system is located in Cellopack company[1], 6^{th} of October city and is used to produce hot air for drying liquid-based ink in huge press, see Figure (1.6). Gas burner is commonly used because of its low cost, and it is essential in case of mass production. In other words, gas burner is essential when the production is continuously produced in 24 hours a day for 30 days a

[1] The project group and supervisor would like to thank Cellopack Company (Cairo-Egypt) for their collaboration in the application.

month (this is the case in Cellopack Company). In such applications, the use of electric heater or UV lambs increases the production cost.

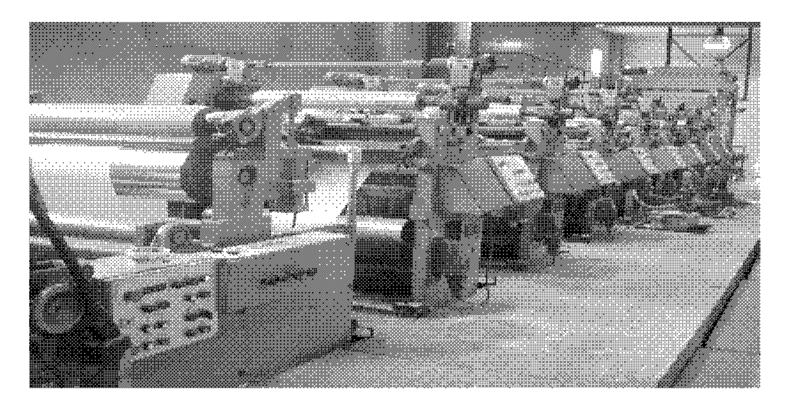


Figure 1.6 Low cost, liquid-based ink printing machine located in Cellopack company, 6th of October city

- **One-Degree of freedom tracking mechanism (Mechanical System):** This is the second application for which the mechanism can track (follow up) a moving object. The nature of something that tracks and the something that is tracked depends on type of application that uses tracking mechanism. Tracking plays an important role in many applications such as:

  (1) Supervisory cams for security systems.

  (2) Sun tracking for renewable energy systems.

  (3) Chaparral missile systems used by air force defense forces.

  The previous appliactions must be automated systems because tracking process is considered as a boring process for human because of the long time and the high accuracy which may be required, so it may need special-skilled person. Therefore, it is preferable to do it automatically. In this prototype, a simple plate has been introduced to simulate the moving objects, a simple web cam is attached to the tracking system to follow and record the moving object.

As shown above, both applications are based on single-degree of freedom controllers. Four types of controllers will be applied for the sake of comparison and studying the performance.

1.4 Thesis Objectives and Structure

The primary objective of the project is to apply, develop, refine and evaluate control systems for the two applications mentioned above. Chapter 2 of the thesis describes these systems in more detail, including consideration of the wider context of the research, hardware and software arrangements, measured variables and formal control objectives.

Chapter 3 discusses Proportional-Integral-Plus (PIP) control design methodology with the help of straightforward examples. The chapter also considers limitations of all applied P-, I-, PI- and PIP approaches.

The two applications are developed in Chapters 4 and 5. In the first instance, the Hot-Air system is used in Chapter 4 to show the first implementation of PIP control methods in Egypt in one of the leading companies. However, Chapter 5 is used to apply the same control systems and evaluate control algorithms, i.e. P-Control, I-Control, PI-Control and PIP control based on the same estimated model.

The conclusions are presented in Chapter 6, which also makes recommendations for future research.

Table of Contents

List of Figures

CHAPTER 2

INTRODUCTION TO DEMONSTRATORS

This chapter introduces the two applications of the thesis, initially focusing on the heat transfer system located on Cellopack Company, 6^{th} of October city. Thereafter, the chapter will describe the mechanical system namely one-degree of freedom tracking mechanism. The two practical demonstrators are single-input single-output SISO system. The first demonstrator is a real full-scale system, however the second is a prototype which can help studying and applying various types of controllers and evaluates their performances.

The control frameworks for these two demonstrators are depicted in Figure (2.1), which shows that online feedback plays a vital part in controlling the process in both cases. However, in the second demonstrator, an additional sensor has been added in order to updating the set point. This project concentrates on the development of improved (low-level) process control for these systems. In the first instance, however, the present chapter introduces the various hardware and software requirements.

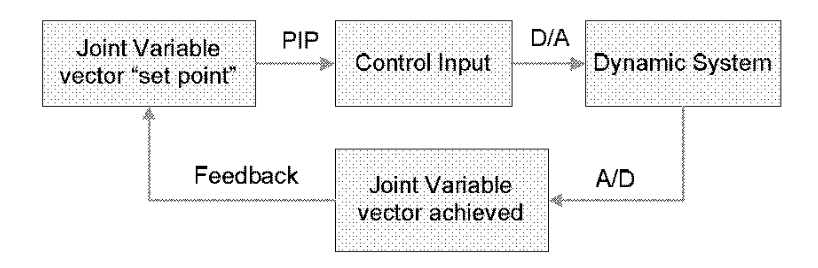


Figure 2.1 General layout of the two demonstrators.

2.1 Hot Air System

The use of hot air in packing industries is very essential. It is used in several applications such as drying low-cost, liquid-based ink in printing machines, see Figure (2.2); also it is used for drying glue existed in carton-pipes. These pipes are used for wrapping sheets of papers, aluminium, nylon, etc. These carton-pipes are made from sheets of scrape papers for which a special type of glue is existed between every two successive sheets. The number of layers depends upon the strength required.

There are several methods for producing hot air. In Cellopack Company, the hot air is produced by electric heaters which used for small machines, Ultra Violet lamps which used for high quality, special types of ink, and gas burner which used for low-cost, liquid-based ink, see Figure (2.2).



Figure 2.2 Low-cost, liquid-based ink printing machine.

Gas burner is commonly used because of its low cost (Natural gas in Egypt is so cheap), and it is essential in case of huge mass production. In other words, gas burner is essential when the production is continuously produced in 24 hours a day for 30 days a month (this is the case in Cellopack Company). In such applications, the use of electric heater or UV lambs increases the production cost.

The existing gas burner in Cellopack Company, see Figure (2.3), is based on natural gas for which the flow rate of the natural gas is controlled by means of mechanical gate. The power of the burner is about 0.5 kilo watts. This gate is motorised by DC motor which is attached to industrial PLC (Programming Logic Control). The hot air is passing through isolated ducts to the target machine to do the required job. At the end of the duct, there is a thermocouple which feedbacks the current temperature to the PLC unit, see Figure (2.3). The role of industrial PLC is to determine the proper opening of the gate such that the temperature measured is around the required (set) temperature.

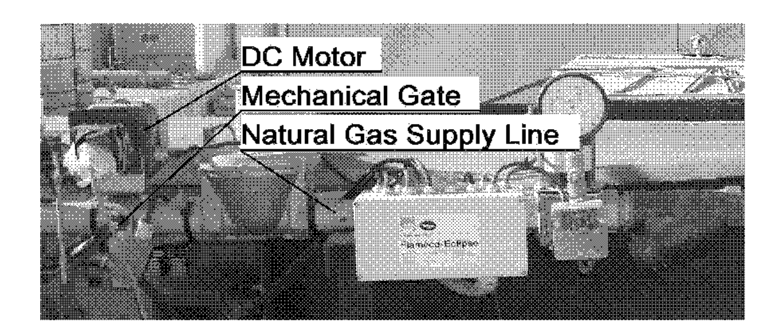


Figure 2.3 Mechanical gate controlling the flow of natural gas into the burner.

The required set temperature ranges from 60 °C to 150 °C depending upon the product material. The performance of the PLC which supplied from the producer of the gas burner is ±10 °C from the set point with settling time about 30 minutes. There is a set point selector which is supplied with the industrial PLC unit, which allows the selection of the required set temperature, see Figure (2.4).

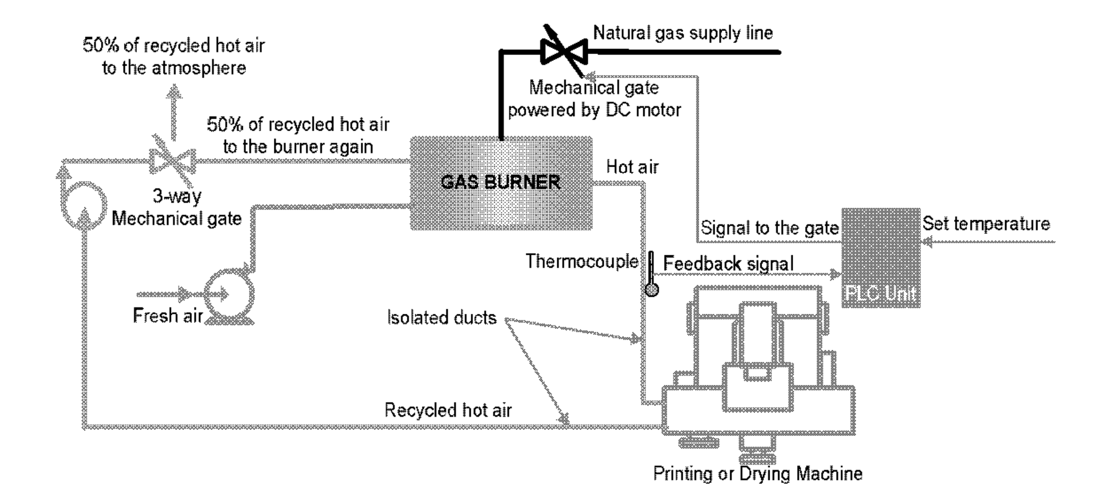


Figure 2.4 The original gas burner control system.

It is not possible to recycle all the hot air again to the gas burner because of two reasons; the first is the need of oxygen from fresh air to keep the flame of the burner on. The quantity of the fresh air needed in order to keep the flame on depends upon the length of the flame and, of course, the length of the flame depends on the opening of the mechanical gate existed in the natural gas supply line, see Figure (2.4).

The second reason is the existence of toxic solvent in the recycled hot air (especially in case of printing machines which use liquid-base ink). This toxic solvent can harm the workers respiratory system and it could be exploded in case the temperature of the hot air reaches its flash point. Therefore, as shown in Figure (2.4), half the amount of the recycled hot air is supplied again to the burner, however the rest of the air quantity required is the fresh air. This helps to economise the use of natural gas as well as reduce the hazards arisen from the toxic solvent in the recycled hot air. This amount of supplied recycled hot air into the burner again shows, to some extent, good enough economic consumption for the natural gas, as well as not very high existence of toxic solvent.

Practically, according to the burner design, when the measured temperature reaches to the set point, the length of the flame of the burner is relatively small and the required amount of fresh air is correspondingly small (this is very important to reduce the

consumption of natural gas). However, for long term use of the hot air, the recycled hot air is full of and saturated with the solvents, therefore, the amount of fresh air should be increased and more than 50% of the recycled hot air should be getting out of the system.

Finally, it should be noted here that there is no control system to control the flow rate of the fresh air. Also, there is no control system to control the 3-way mechanical gate to control to amount of exhausting recycled hot air.

2.1.1 Problem statement

Since the existing industrial PLC unit of the gas burner is broken, it is replaced by two push buttons, up and down which manually open and close the natural gas gate, see Figure (2.5). An expertise worker is required now in order to use this manual procedure. The performance of this manually-controlled system is $\pm 25\ ^{\circ}\mathrm{C}$ which is not acceptable at all. Moreover, an ON-OFF switch is added to switch the blower of fresh air ON in the beginning of heating and then OFF when the current temperature of the hot air reaches around the set point, this is to economise the use of natural gas. Finally, the 3-way mechanical gate for exhausting the recycled hot air is fixed at 50% of the recycled hot air is exhausting outside the system; however the rest 50% is supplied to gas burner again.

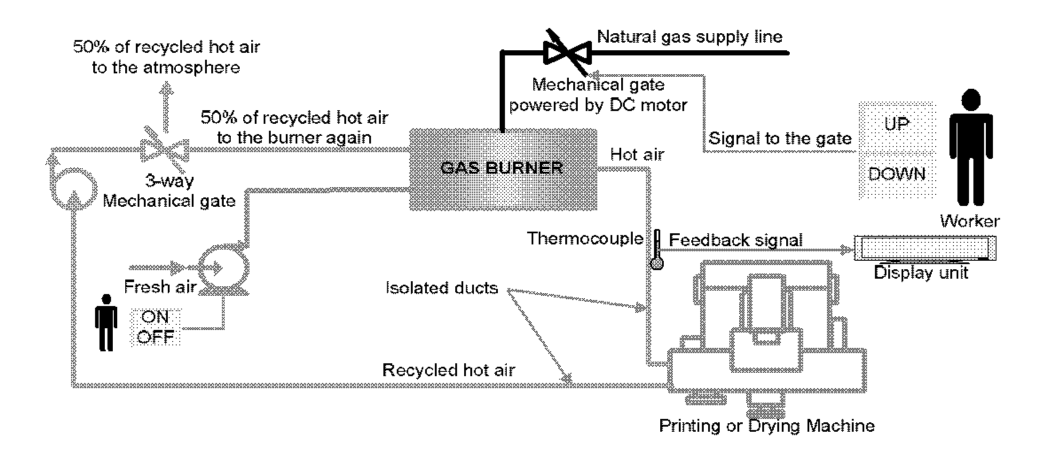


Figure 2.5 The manually-controlled gas burner.

The target control system is automatically control the opening of the natural gas gate in order to set the hot air temperature to the required level according to the worker selection. The performance of this stage should be $\pm 5\,^{\circ}\mathrm{C}$ with settling time no more than 30 minutes. This should be done regardless the state of the fresh air blower (ON or OFF). Here, the system is SISO (1 input and 1 output). The input is the position of the natural gas gate, and the output is the temperature of the hot air.

2.1.2 Hardware/Software

1- PT100

In order to get the above achievement, a precision platinum temperature sensor "PT100" has been added at the end of the duct carrying the hot air, just before the machine, to measure the temperature of the supplied hot air. It offers excellent accuracy over a wide temperature range, from $-200^{\circ}\mathrm{C}$ to $+850^{\circ}\mathrm{C}$. Moreover, it is easy to install and can be used for industrial application for its reliability. PT100 is available from many manufactures to suite various applications.

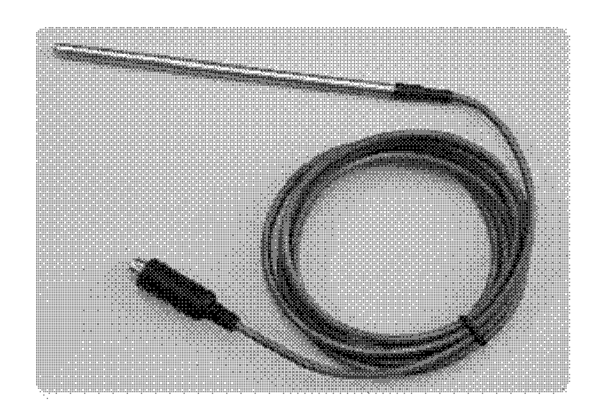
The principle of operation is to measure the resistance of a platinum element. The most common type of PT100 has a resistance of 100Ω at $0^{\circ}\mathrm{C}$ and 138.4Ω at $100^{\circ}\mathrm{C}$. The relationship between temperature and resistance is approximately linear over a small temperature range. For example, by assuming that PT100 is linear over the range $0-100^{\circ}\mathrm{C}$, the error at $50^{\circ}\mathrm{C}$ is $0.4^{\circ}\mathrm{C}$.

For more precise measurement, it is necessary to linearise the resistance. The most recent definition of the relationship between resistance and temperature is the International Temperature Standard 90 (ITS-90), the linearisation can be done automatically by using a signal conditioner that correct the resistance of the PT100 according to the following equation[1],

$$R_c = R_{\mathrm{PT100}}\left(1 + A\,t + B\,t^2 + C\,(t-100)\,t^3\right) \tag{2.1}$$

[1] **http://www.picotech.com/applications/pt100.html.**

for which R_c is the corrected resistance and R_{PT100} is the resistance from PT100, both are in $m\Omega$. *A*, *B* and *C* are constants for which $A = 3.9083\times10^{-3}$, $B = -5.775\times10^{-7}$ and $C = -4.183\times10^{-12}$ if $t \leq 0^{\circ}C$ or $C = 0$ if $t > 0^{\circ}C$. From equation (2.1), it is clear that $1^{O}C$ temperature change requires adding a resistance correction 0.384Ω. The linearisation procedure discussed in equation (2.1) can be done by means of microprocessor based temperature transmitter or VT-NB board from Vertex[2], see Figure (2.7). Also, its dimensions can be depicted in Figure (2.8).



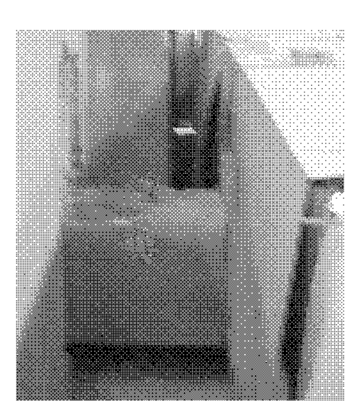


Figure 2.6 Left: The precision platinum temperature sensor "PT100",

Right: The location of PT100.

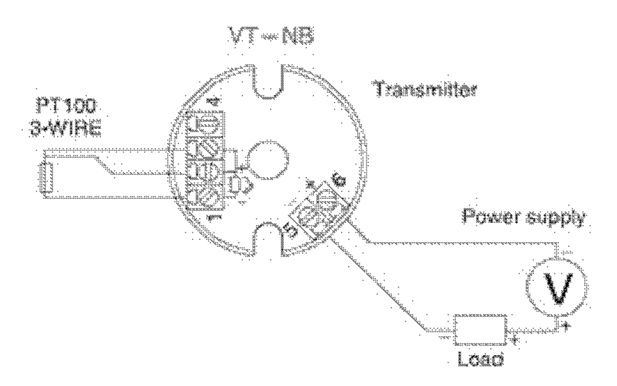


Figure 2.7 VT-NB board for excitation and linearisation of PT100

[2] **http://www.vertex-tw.com/english/nbnr.htm**

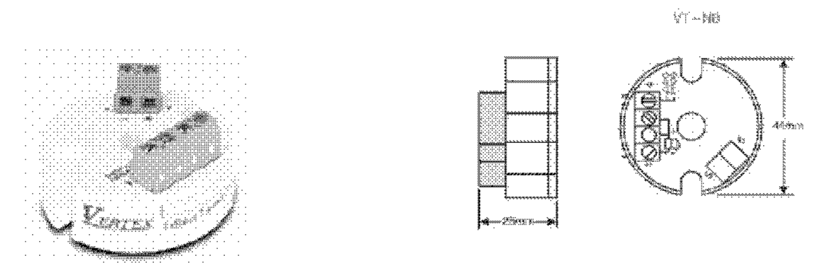


Figure 2.8 Mechanical properties of VT-NB board

Finally, Since the duct carrying the hot air has a square cross sectional shape with approximately 40cm side length, the length of PT100 has been selected to be 20cm, in order to measure the temperature of the core of the hot air flow.

2- Absolute encoder

Also, an absolute encoder with brand "HONTKO"[3] has been added and installed at the centre of rotation of the gate of the natural gas to feedback the current opening of the gate. This encoder gives 256 pulse/revolution. In other words, it can measure angle from 0° to 360° in digital form from 0 to 255 digits with response 10kHz. Therefore, the encoder can divide the 360° into 255 levels for which each level measures $360/255 = 1.4^{\circ}$. The maximum opening for the natural gas gate is approximately 20° which, therefore, can be normalized from $0 \rightarrow 13$. In other words, actuator input $u = 0$ corresponds to no opening for the gate; however actuator input $u = 13$ corresponds to full gate opening. The output of the encoder is binary reflected gray code.

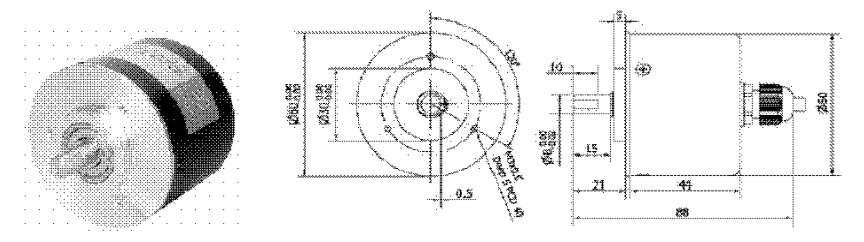


Figure 2.9 HONTKO absolute encoder with its mechanical specification

[3] **www.HONTKO.com.tw.**

Gray codes are used in position encoders to avoid the possibility that, when several bits change in the binary representation of an angle, a misread could result from some of the bits changing before others. Rotary encoders benefit from the cyclic nature of Gray codes because the first and last values of the sequence differ by only one bit[4]. There is an algorithm to change from Gray code to Binary code as follows:

1- Write down the number in Gray code.
2- The msb of the Binary code is the msb of the Gray code.
3- Add the last obtained Binary bit to the next significant bit of the Gray code number to get the next Binary bit.
4- Repeat step "3" till all the bits of the Gray coded number have been added. The resultant number is the Binary equivalent of the Gray number.

For example, the Gray code number 1101101 can be converted to Binary equivalent xxxxxxxx as follows:

1- The msb in the Binary equivalent will be the same as the msb in the Gray code, therefore the Binary will be 1xxxxxxx.
2- The second Binary bit can be obtained by adding the last obtained Binary bit "1" to the second Gray bit "1". This gives "0"; therefore, the Binary will be 10xxxxxx.
3- The third binary bit can be obtained by adding the last obtained Binary bit "0" to the third gray bit "0". This gives "0"; therefore, the Binary will be 100xxxxx.
4- Repeating the last step, the resulting Binary will be 1001001.

---

[4] **http://en.wikipedia.org/wiki/Gray_code.**

The above algorithm can be coded as follows,

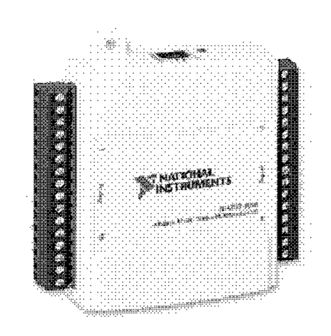
```
Let    G [n : 0]    be the input array of bits in Gray code
Let    B [n : 0]    be the output array of bits in the usual Binary representation
B [n] = G [n]
for i = n – 1 down to 0
       B [i] = B [i + 1] XOR G [i]
end
```

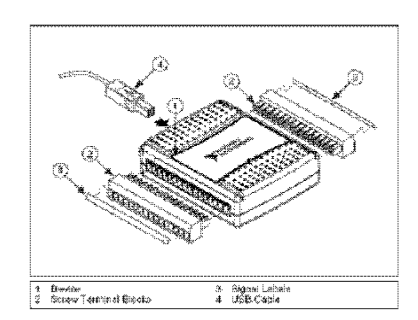
Finally, it should be noted here that the rate of change of the normalised input, u, is $\dot{u} = 0.25/\sec$. In other words, the encoder reading shows 1 incremental change every 4 seconds. This hardware restriction of the gate opening should be taken into consideration during sampling rate selection as will be shown later.

3- NI 6008 DAQ

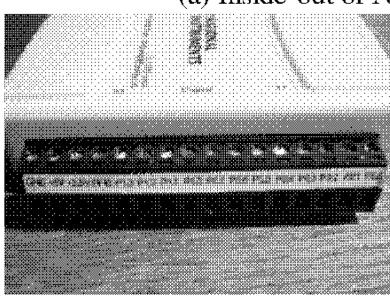
The NI 6008 USB DAQ is a USB data acquisition and control device with analogue I/O and digital I/O, see Figure (2.10), it is used to interface the peripherals (sensors and actuators) with a PC. Its main specifications are:

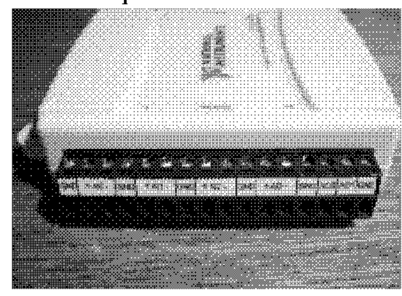
- 8 analogue inputs with referenced single ended signal coupling, with 12 bit A/D converter and sampling rate 10 kS/s.
- 2 analogue outputs with voltage range $0-5\,\mathrm{V}$ (fixed) and 12 bit A/D converter. The output rate is $150\mathrm{S/s}$.
- 12 digital I/O channels which can be configured individually.
- 32 bit counter.
- On-board voltage sources available at individual terminals, 2.5 V and 5.0 V.
- NI 6008 DAQ is powered via the USB cable, and is compatible with LabVIEW® or Visual Studio®. Also, it can work under Windows, Mac and Linux.





(a) Inside-out of NI 6008 USB Data Acquisition





(b) Digital ports (left) and analogue ports (right) for NI 6008 USB Data Acquisition

Figure 2.10 NI 6008 USB Data Acquisition

Since the DAQ hardware is compatible with LabVIEW®, therefore it is connected to Laptop for which LabVIEW®, release 8.2, has been installed. The specification of the Laptop is Core2Duo with 3 MB cash memory, 1 GB RAM, and 160 GB HDD.

8 digital I/O ports is used as digital inputs for the encoder and 2 digital I/O ports are used as digital output for the two-relays as up and down for the natural gas gate. 2 analogue inputs are occupied by VT-NB board which connected to the temperature sensor PT100 to measure the voltage difference rather referenced voltage to minimise the noise. Also, 2 analogue inputs are used to get the required set temperature from the set point selector. Here, 2 input analogue ports are used to get the difference of voltage rather than referenced voltage to minimise the noise. Finally, 1 analogue output is used for displaying the current temperature on the set point selector LCD, see Figure (2.11).

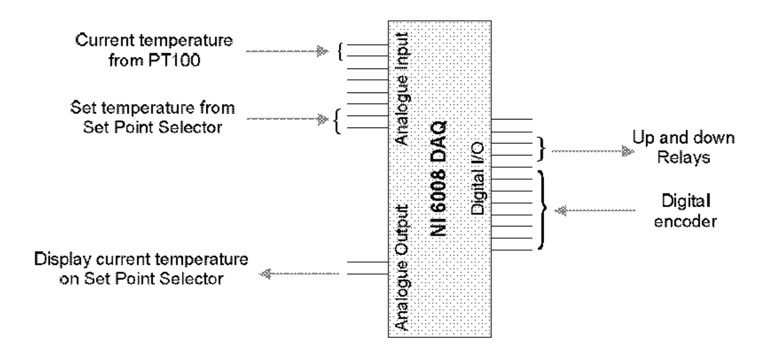
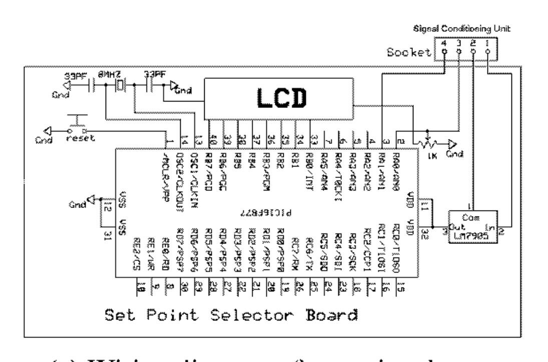


Figure 2.11 Schematic shows the occupied ports on the NI 6008 USB DAQ

<u>4- Set point selector</u>

In stead of using the keyboard to select the set point of temperature, Set Point Selector unit has been assembled and integrated with the system to facilitate the use of the new controller, see Figure (2.12).



(a) Wiring diagram of set point elector



(b) Snap shot of set point selector

Figure 2.12 Set Point Selector

<u>5- Signal conditioning unit</u>

In order to connect the peripherals to DAQ and then to PC, it is required to recondition the received or transmitted signals. The wiring diagram of the conditioning unit is

depicted in Figure (2.13); however a snapshot for its location with the PC can be shown in Figure (2.14).

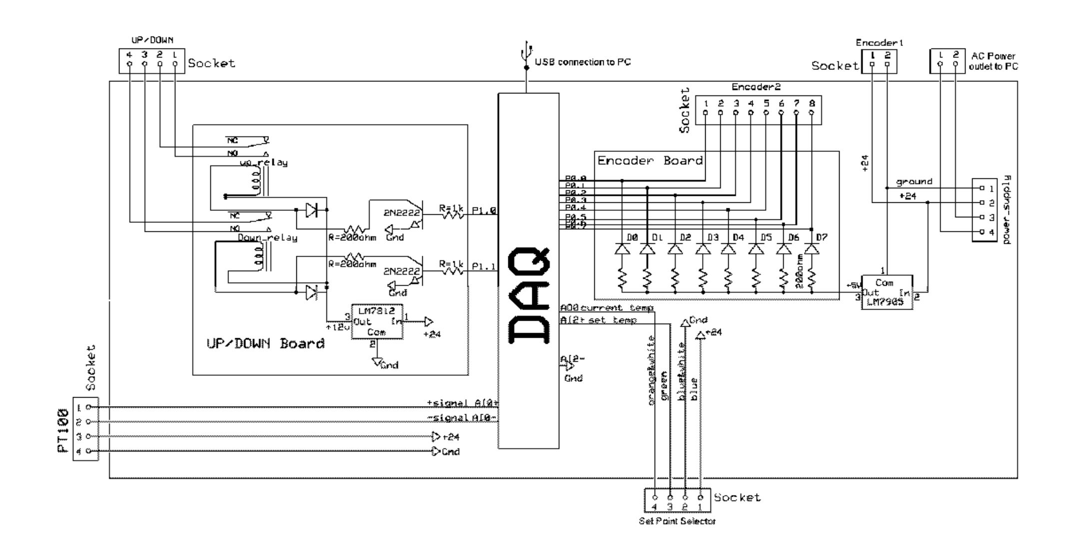


Figure 2.13 Wiring diagram of Signal Conditioning Unit

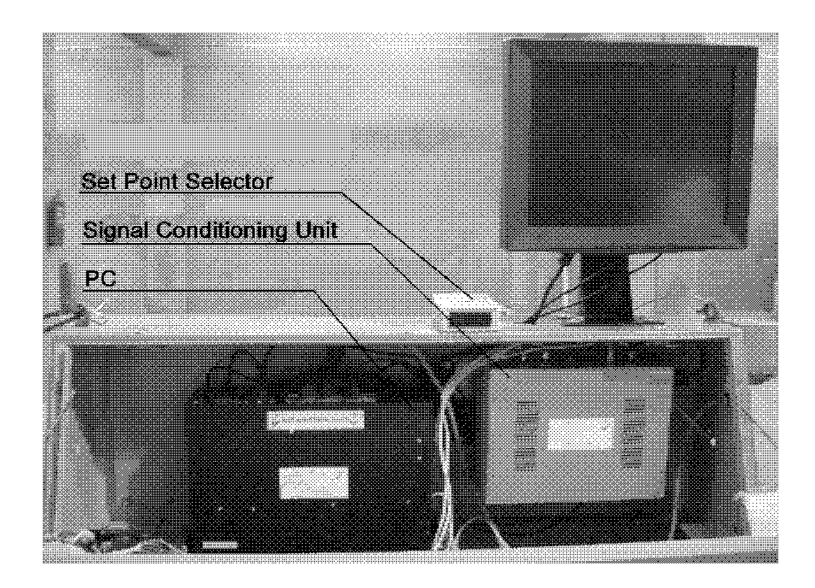
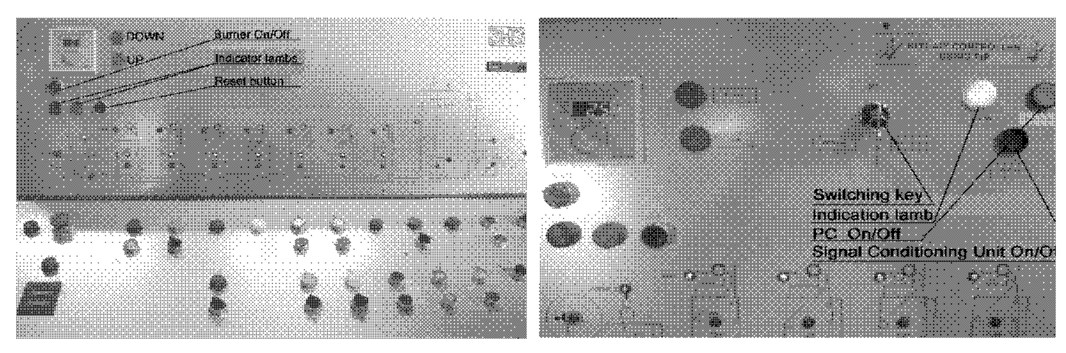


Figure 2.14 The location of the Signal Conditioning Unit with the PC

6- Switching between manual and automatic modes

Originally, the manual control system was based on Up/Down buttons, see figure (15-a). A switching key has been added to the system to switch between manual and automatic modes, see Figure (15-b). The switching procedure is based on activating the actuator through the manual Up/Down bush buttons or activating the actuator through automatic Up/Down relays existing in the signal conditioning unit, see Figure (2.16). This is done by means of break one of the two circuits via the common line. This helps avoiding interference which could happen between the two modes.



(a) Before adding automatic controller (b) After adding automatic controller

Figure 2.15 (a) The Up/Down (manual) control system and
(b) The automatic control system

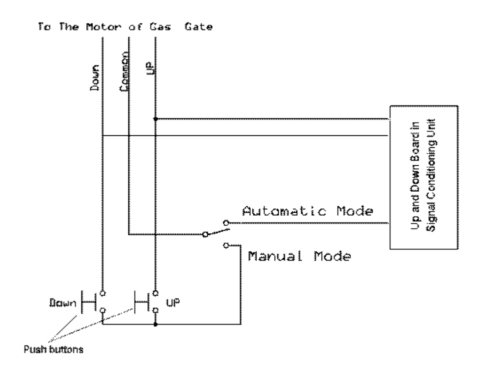


Figure 2.16 Wiring diagram shows the switching between
manual and automatic modes

2.1.2 Hot air system calibration

1- PT 100:

Calibration of PT100 has been carried using simple thermometer for which several voltages has been measured by means of PT100 with its corresponding temperature. The equation that best relates the measured voltages from PT100 with its corresponding temperatures is found to be

$$y = round\ (118.16\ V - 106.39) \tag{2.2}$$

The above equation can be described by means of visual programming of LabView® as shown in Figure (2.17).

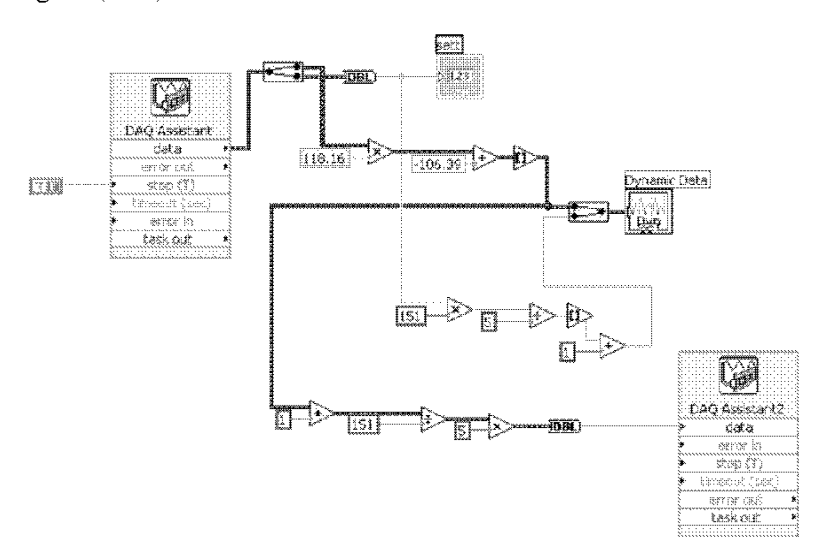


Figure 2.17 Visual programming of equation (2.2) using LabView®

2- Absolute encoder:

Since the encoder gives 256 pulse/revolution, then it can divide the 360° into 255 levels for which each level measures $360/255 = 1.4^\circ$. The maximum opening for the natural

gas gate is approximately 20° which, therefore, can be normalized from $0 \rightarrow 13$. In other words, actuator input $u = 0$ corresponds to no opening for the gate; however actuator input $u = 13$ corresponds to full gate opening. The output of the encoder is binary reflected gray code. Several trials has been carried out to calibrate the encoder, such that the encoder reads 0 at no opening and it is found that the equation that best describe this behaviour is:

$$u = -round\left(\frac{\text{binary code}}{255} \times 100\right) + 4 \tag{2.3}$$

The above equation can be described by means of visual programming of LabView® as shown in Figure (2.18).

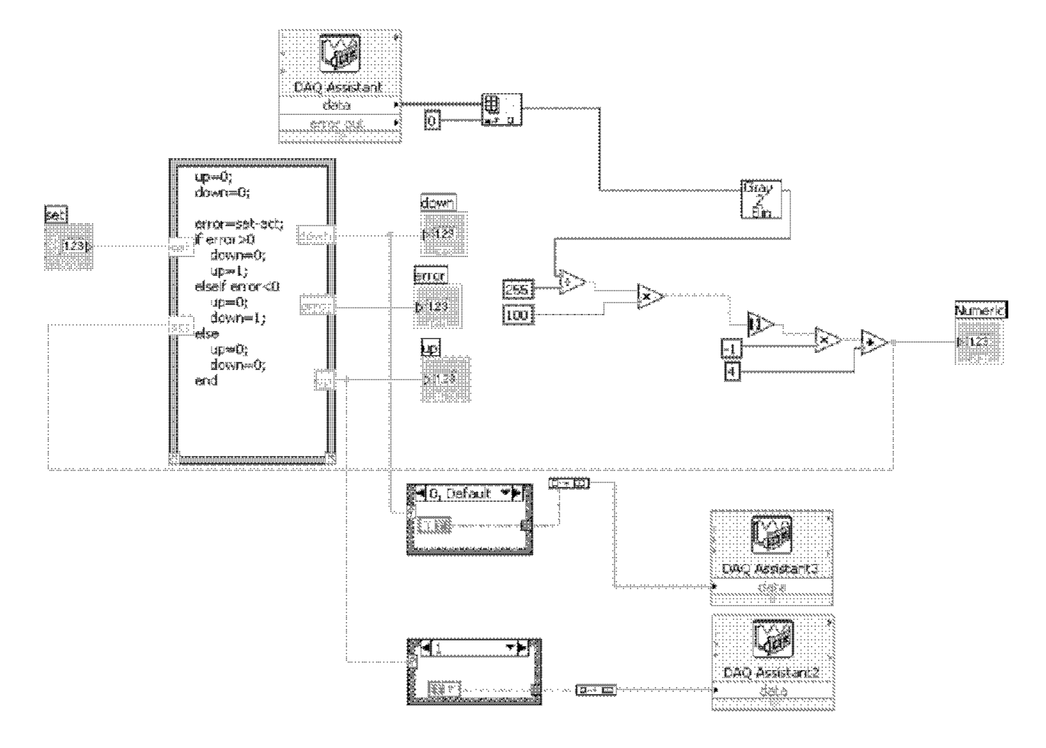


Figure 2.18 Visual programming of equation (2.3) using LabView®

2.2 One-Degree of Freedom Tracking System

The second practical example is the automation tracking mechanism. *Tracking* is the process when we want to follow up the objective trajectory. The process of tracking can be considered as a boring and hazardous process since it may last considerable time and the tracked object could be dangerous. Moreover, automization of tracking process increase the accuracy of the image of the tracked object. A single-degree of freedom has been chosen for this application because the objective of the example is demonstrating several digital controllers rather than just track an object. A simple plate has been chosen to be tracked, however, in real life, the tracked object could be any moving element.

There are several applications for automatic tracking, for example, it can be used in security. In other words, one tracking cam can take place several cams. In zoology and entomology fields, tracking cam can be used to track animals or insects to know more about their life. Also, moving or tracking cam can be used in production line observation. Moreover, in air force defense, tracking strange objects is essential.

2.2.1 Construction of tracking mechanism

The main body of the prototype consists of a base which made from Aluminum, toothed belt to work as a conveyor to move the cam up/down, see Figure (2.19). A simple plate has been added to replace and simulate the action of the moving object which required to be tracked. A DC motor has been used as an actuator and a simple Nickel-Chrome wire has been used as a potentiometer to feedback the current position of the tracker or cam. Finally, one more potentiometer is used to feedback the set point of the moving plate. This last potentiometer can be easily taken out and replaced by infra sensor or the tracked object can be tracked visually using the cam.

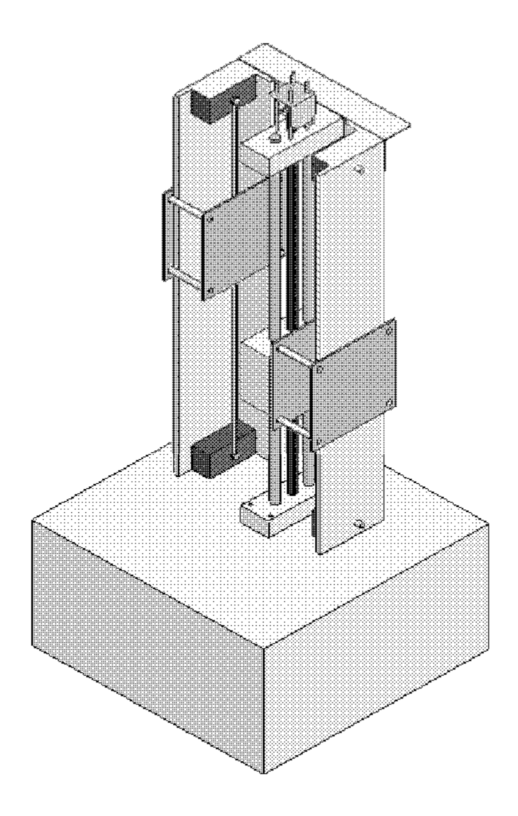


Figure 2.19 3-D drawing of the tracking mechanism protptype

2.2.2 Hardware components

1- DC motor 24 volt

A DC motor works by converting electric power into mechanical, so actuating the sliding joint, see Figure (2.20). DC motor has been selected for the following reasons:

- Low initial cost.
- High reliability.
- Simple control of motor speed.

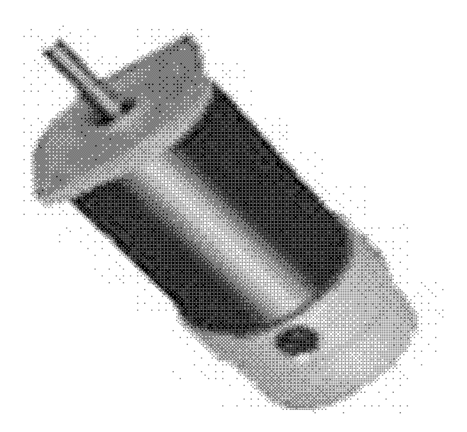


Figure 2.20 DC motor

2- Potentiometer

Two potentiometers have been designed and produced. The first is used to feedback the current position of the tracker or cam and the other is used for determining the current position of the tracked object (the simple plate). They made from Nickel-Chrome wire and a pointer to change the resistance and so they can feedback a part of a voltage which corresponds to the current position of the pointer.

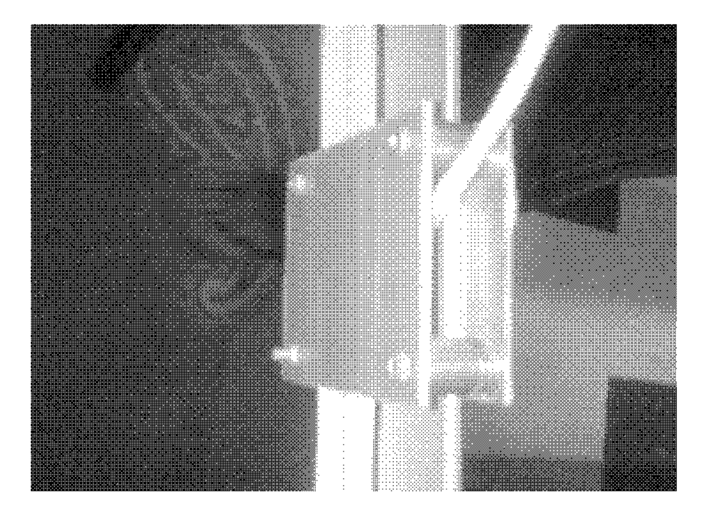


Figure 2.21 The pointer of the potentiometer

A DC voltage source with 5.7 volt is connected to the Nickel-Chrome wire, so by moving the pointer, the voltage will change according to the position of the pointer. The wiring diagram of the potentiometer is depicted in Figure (2.22). It should be noted here that the wire is made from Nickel-Chrome for the following reasons:

- High accuracy
- Low heat effect
- Better response

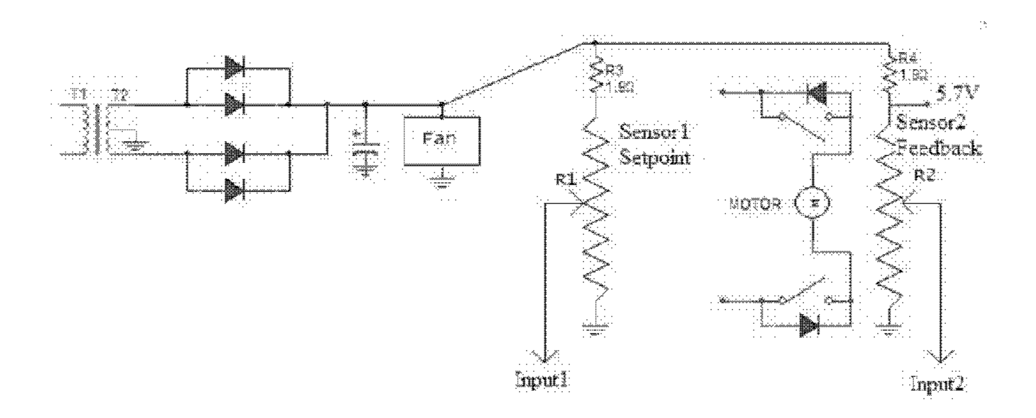


Figure 2.22 Wiring diagram of the potentiometer

The available potentiometers in the market are not suitable for this application. Therefore, the two potentiometers have been designed and produced using local materials. However, a considerable noise is noticed, it is found that these noise helps to evaluate the applied controllers.

3- NI 6008 DAQ

As explained before, NI 6008 USB DAQ is a USB data acquisition and control device with analogue I/O and digital I/O, see Figure (2.10).

One digital I/O port is used as digital output for the motor direction, 2 analogue inputs which connected to feedback sensor and set point sensor and 1 analogue output which connected to dimmer.

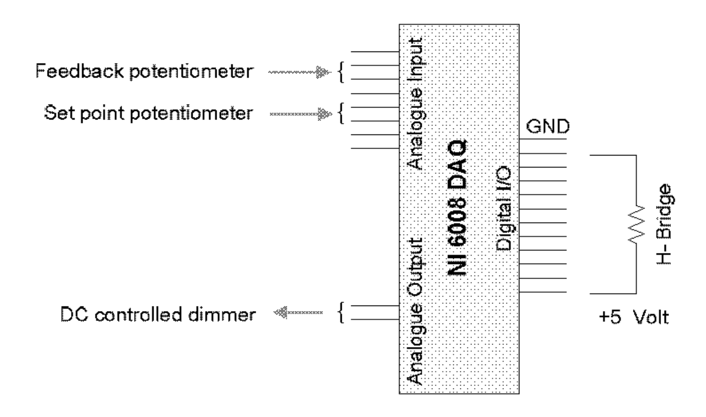


Figure 2.23 Schematic shows the occupied ports on the NI 6008 USB DAQ

4- H-Bridge circuit:

It consists of two relays each 12 Volt, diode 1 A, resistance 10 kΩ and transistor NPN-2N2222, it is used for the direction selection

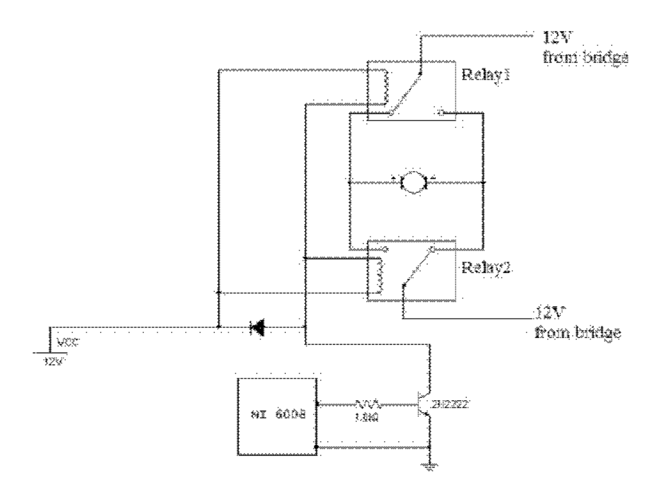


Figure 2.24 Wire diagram of the H-Bridge

5- DC controlled dimmer:

The main function of the dimmer is controlling the speed of the DC motor. Its specifications are:

- Control voltage ranges from $0 \rightarrow 10$ Volt.
- Power supply with 220 Volt, and AC ranges from $50 \rightarrow 60$ Hz.
- Max load is 3.5 A.

It consists of:

1- Resistors with $R_1 = 4$ kΩ, $R_2 = R_3 = 100$ kΩ, $R_4 = 470$ kΩ, $R_5 = 1$ MΩ, $R_6 = 220$ kΩ, and $R_7 = 15$ kΩ.

2- Diodes with D_1 and D_2 :1N4148, D_3 :1N4007.

3- Capacitors with $C_1 = 4$ nF, $C_2 = 100$ μF, $C_3 = 100$ μF/250 VAC, $C_4 = 100$ nF.

4- Potentiometers with $RV_1 = 220$ kΩ, $RV_2 = 2$ MΩ

5- TR_1 : TRIAC.

6- Coil $L_1 = 50$ μH/6 A

7- IC's have the following types IC_1 :4N27, IC_2 :TEA1007

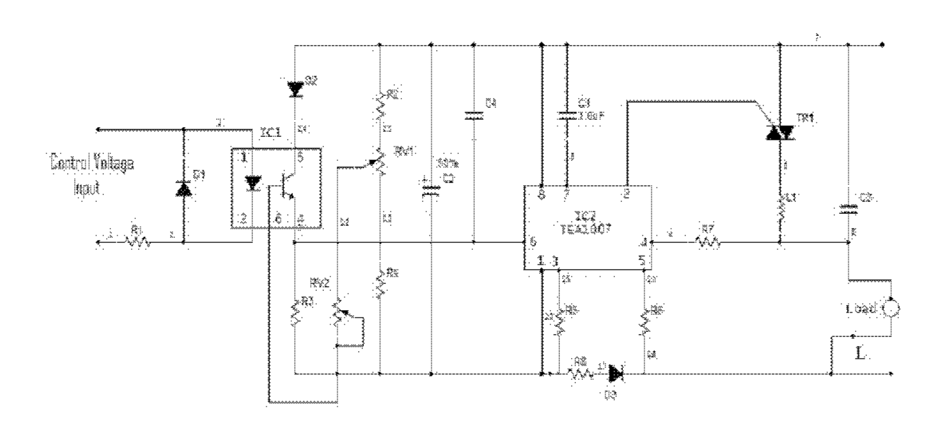


Figure 2.25 Wire diagram of the dimmer

6- Circuit Integration:

The wiring diagram of integrated all the above circuits in Figures (2.22, 2.23, 2.24 and 2.25) is depicted in Figure (2.26). However, a snapshot of the control and circuit conditioning unit can be shown in Figure (2.27).

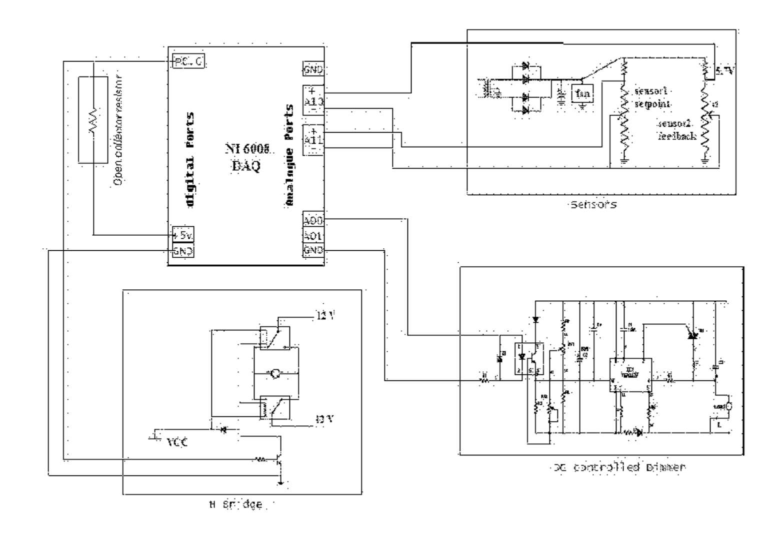


Figure 2.26 Wiring diagram of the all circuits

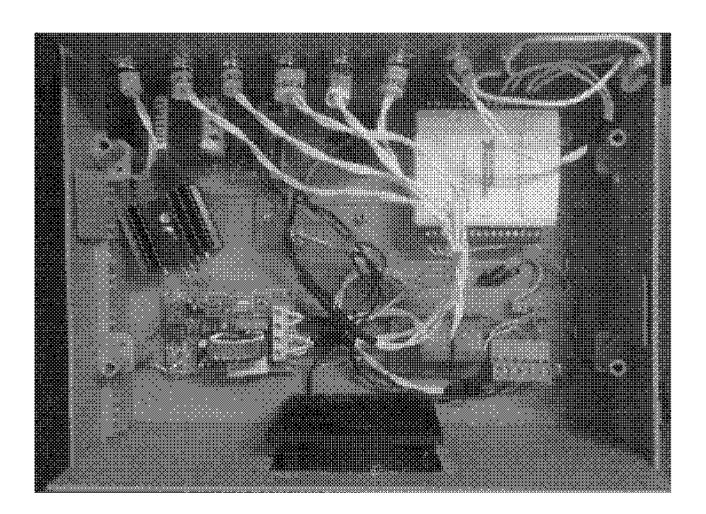


Figure 2.27 Snapshot of the control and signal conditioning box

2.2.3 Tracking system calibration

1- Potentiometer:

Calibration of potentiometer has been carried using simple ruler for which several voltages has been measured by means of potentiometer with its corresponding height. The equation that best relates the measured voltages from potentiometer with its corresponding plate height is found to be

$$y = -27\ V_y + 83.375 \tag{2.4}$$

for which V_y is the measured voltage and y is the measured distance by using a simple ruler. It is convenient to note that equation (2.4) is suitable for both potentiometers.

2- DC motor calibration:

Normalizing the input voltage is essential in this case since the motor receive only the absolute voltage, whilst the relays are responsible for the sign of the input voltage. Therefore, for modeling purposes, the input voltage is normalized to lie in the range -1000 to $+1000$. Here, negative and positive values imply that the cam-conveyor system moves downward or upward respectively. Also, a normalized voltage of zero is chosen to correspond to the ValveMid. The analogue value for the ValveMid[5] setting is known to be exactly 0. This corresponds to a zero analogue input A trial and error procedure is carried out by means of LabView® program to define the deadband limits, ValveDeadbandLo[6] and ValveDeadbandHi[7], these values are found to be:

[5] **ValveMid** is the of value of the analogue voltage for which the motor receives no input.
[6] **ValveDeadbandLo** is the value of the input analogue voltage for which starts to overcome the static friction and moves downward.
[7] **ValveDeadbandHo** is the value of the input analogue voltage for which starts to overcome the static friction and moves upward

$$\begin{aligned} \text{ValveMid} &= 0.0 \ \text{Volt} \\ \text{ValveDeadbandLo} &= 2.7 \ \text{Volt} \\ \text{ValveDeadbandHi} &= 3.1 \ \text{Volt} \end{aligned} \tag{2.5}$$

Finally, from Figure (2.28), the mapping equation between the normalized input, u, and the digital input is as follows,

$$u = \begin{cases} \text{ValveDeadbandHi} + \text{Demand} \times \dfrac{\text{ValveMax - ValveDeadbandHi}}{1000} & \text{if } u > 0 \\ \text{ValveDeadbandLo} - \text{Demand} \times \dfrac{\text{ValveMin} - \text{ValveDeadbandLo}}{1000} & \text{if } u < 0 \end{cases} \tag{2.6}$$

for which ValveMax is the maximum analogue voltage can be sent to the DC motor. In this case, both ValveMax and ValveMin are equal to $5 \ \text{Volt}$.

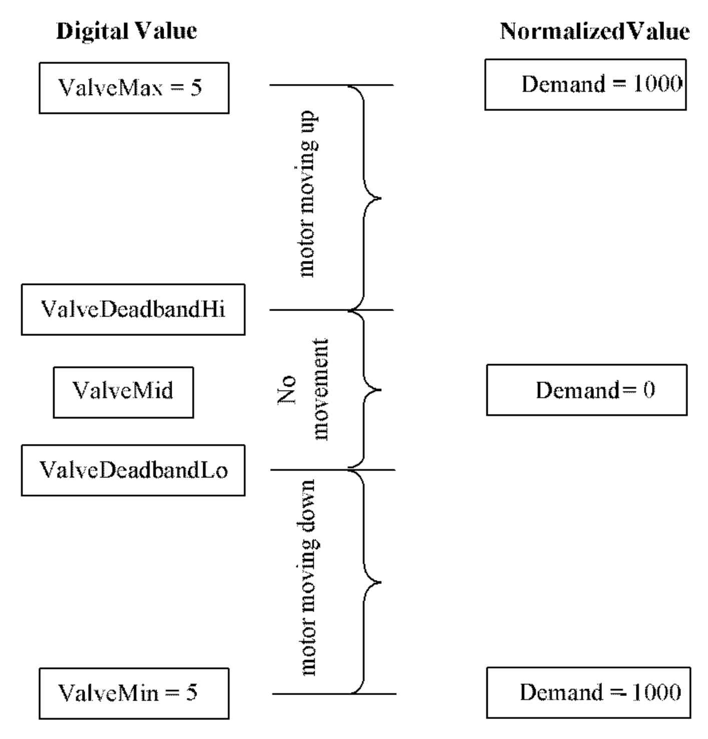


Figure 2.28 Schematic diagram showing the normalization process for the DC motor

2.3 Software Interface

In order to achieve the control aims for both demonstrators, it is necessary to build programs capable of interfacing the proposed control algorithm with the sensors and the actuators. In this regard, LabView® has been selected since it is compatible with the NI 6008DAQ, also LabView® is now becoming one of the leading program not only in the laboratories but also in industry.

2.3.1 Program for the hot air system

The LabView® program for the hot air system allows the user to input the required temperature. Also, a complete supervisory control is shown through the front panel of the program, see Figure (2.29).

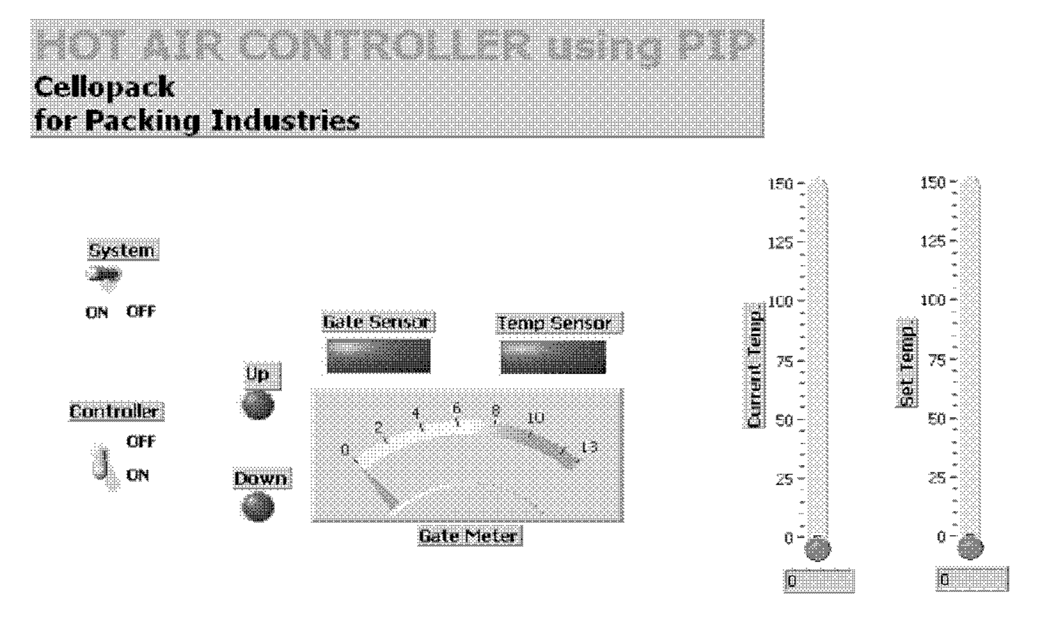


Figure 2.29 The front panel for the LabView® program for hot air system

2.3.2 Program for the tracking system

In this system, the LabView® program is just used to supervise the action and see the performance since the set point is automatically detected through the potentiometer.The complete front panel is shown in Figure (2.30).

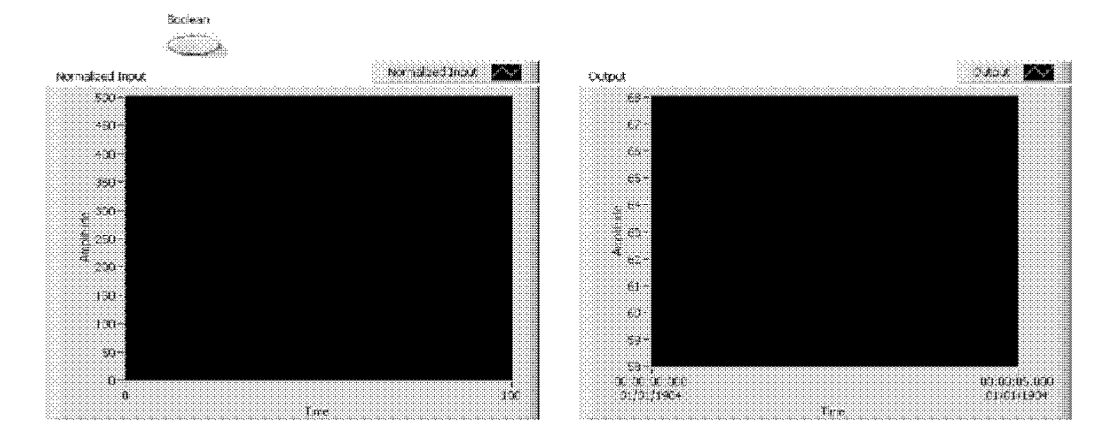


Figure 2.30 The front panel for the LabView® program for tracking system

Table of Contents

List of Figures

CHAPTER 3

PIP CONTROL SYSTEM DESIGN

Proportional Integral Plus (PIP) control is a modern conventional control which is based on True Digital Control (TDC). PIP can be interpreted as a logical extension of conventional PI controller with additional dynamic feedback and input compensators introduced automatically when the process has second order or higher dynamics, or pure time delays greater than unity sample. This chapter briefly describes the PIP control algorithm, in order to help the reader to understand the servomechanism of the approach.

The first stage of PIP control system design is to identify and estimate appropriate discrete-time transfer function (TF) model. Here, both equation error and response error methods are utilised, to identify and estimate the TF model, before turning to control system design.

3.1 Dynamic Modelling (Identification and Estimation)

The present work utilises two statistical stages for discrete-time TF estimation:

1- Identification or determination of the shape of the TF..

2- Statistically estimation of the constant parameters.

Experimental or data based modeling consists of the development of mathematical models of dynamic systems on the basis of measured data from laboratory/field experiments. The data are utilized to both *identify* the model structure (i.e. the orders of the polynomials in a transfer function model and the size of the pure time delay) and to *estimate* the parameters which characterize the dynamic behaviour of the system. Sophisticated *time-series* methods are required for this. So, neither the underlying physics nor internal life of the system need necessarily play a role in model generation. In contrast to physical modeling there are procedures for experimental modeling in which the modeling can be wholly or partially automated.

The difficulties of this approach are numerical in nature since numerical oscillation may occur at abrupt changes. This may impair or even prevent the convergence of the solution. However, a possible solution is offered by procedures that smooth the dynamic system characteristics.

3.1.1 Discrete time transfer functions

If continuous-time data from the system are sampled at a sampling interval of Δt seconds, for simplicity, assume that the measurements start at time $t = 0$, then the variable $y(t)$ can be represented at each sampling point as follows: $y(0)$, $y(\Delta t)$, $y(2\Delta t)$, ..., $y(N\Delta t)$ where N is the total number of samples. Alternatively, it is

possible to use the following subscript notation: y_0, y_1, y_2, …, y_k, …, y_N where y_k is the value of variable at k^{th} sample.

From the previous introduction, the transfer function in discrete-time is simply describe the transfer of the *input* at k^{th} sample u_k to the *output* at k^{th} sample y_k, as shown in Figure (3.1).

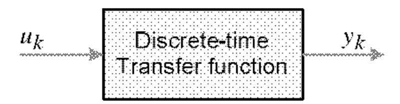


Figure 3.1 Transfer function (TF) in discrete form

For example, if the transfer function is a scalar g (i.e. the TF is a simple gain element), then $y_k = g\,u_k$. The same rules of block diagram manipulation are applied here.

For simpler manipulation of discrete-time TF, it is important to introduce the so called *backward shift operator* z^{-i}, for which

$$\mathrm{z}^{-i}\,y_k = y_{k-i} \tag{3.1}$$

It is clear that z^{-i} introduces a *delay* of i samples. For example, the z^{-1} introduces a delay of one sample (i.e. $\mathrm{z}^{-1}\,y_k = y_{k-1}$).

In order to understand the previous discussion, consider a simple model for the volume of water in a reservoir which has the following incremental form,

$$y_k - a\,y_{k-1} = b\,u_{k-1} \tag{3.2}$$

By using backward shift operator, the above equation can rewritten as

$$y_k + a\,z^{-1} y_k = b\,z^{-1}\,u_k \tag{3.3}$$

Straightforward algebra can be used to write the following transfer function

$$y_k = \frac{b\,z^{-1}}{1 + a\,z^{-1}} u_k \tag{3.4}$$

Alternatively, this can be modeled in block diagram as shown in Figure (3.2),

$$u_k \rightarrow \boxed{\frac{b\,z^{-1}}{1 + a\,z^{-1}}} \rightarrow y_k$$

Figure 3.2 Block diagram representation of system example

for which u_k is the system input (effective inlet flow rate), y_k is the system output (water depth in the reservoir), and the constants a and b are the model parameters required to be estimated.

The simulation of this model can be performed using MATLAB/SIMULINK®, the response is shown in Figure (3.2), assuming that $a = -0.8$ and $b = 0.2$.

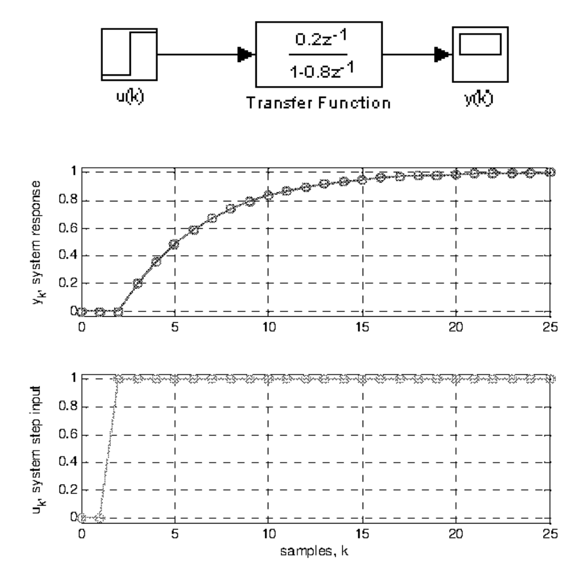


Figure 3.3 Unit step response of the first order system, at $a = -0.8$ and $b = 0.2$

It should be noted here that the above response shows very close dynamics to the first order differential equation of the same system. The difference here is that the reservoir level is only calculated at discrete samples in time, shown as “circles” in Figure (3.3).In more general terms a and b in the discrete equation are unknowns and are statically estimated from time series data as will be discussed later.

3.1.2 The general discrete-time transfer function model

In some cases, as above, first order difference equations are sufficient to model a given system. However, in many cases, it was found that higher order equations are required to represent the dynamic behaviour of the uncontrolled system. For example, second order (or higher) are required to describe oscillatory behaviour. Therefore, attention should be taken to a discrete-time TF representation. For a linear Single-Input, Single-Output (SISO) system, the TF in terms of the z^{-1} operator takes the following form,

$$y_k = \frac{b_o + b_1 z^{-1} + b_2 z^{-2} + \ldots + b_m z^{-m}}{1 + a_1 z^{-1} + a_2 z^{-2} + \ldots a_n z^{-n}} u_k = \frac{B(z^{-1})}{A(z^{-1})} u_k \tag{3.5}$$

where $A(z^{-1})$ and $B(z^{-1})$ are appropriately defined polynomials in the backward shift operator z^{-1}.

It is very important to define some terms. The first term is the steady state response which is the response of the system at $t \to \infty$. In discrete-time case, it is the response at $k \to \infty$ (y_∞) for which the system reaches its equilibrium level and no changes in the response can be noticed. Therefore, in such cases, at steady state response the following criteria should be applied $y_k = y_{k-1}$ [1]. This means that at steady state case, the backward shift operator may be substituted by unity, i.e. $z^{-1} = 1$. Therefore, for the above example, the steady state response can be calculated by replacing $z^{-1} = 1$, i.e

[1] Since there is no change in the response (output) of the dynamic system.

$$y_\infty = \frac{b}{1+a} u_\infty = 1 \tag{3.6}$$

Another term is the steady state gain which is the numerical value that relates the input u_∞ and the output y_∞. It can be obtained by replacing the backward shift operator z^{-1} by unity. Therefore, in the same example, the steady state gain is $G = b/(1+a)$. For the same system example $G = 0.2/(1-0.8) = 1$. This means that the output y_k asymptotically converges to the steady state value of unity which is the same magnitude as the unit step input, see Figure (3.3).

One of the most important terms which requires to be defined is the time delay δ. This term is defined as the number of samples required for the system to start respond to the input. For the above example, the time delay is $\delta = 1$. However, if the polynomial $B(z^{-1})$ starts at z^{-p}, then the time delay $\delta = p$.

Finally, the order of the TF can be defined by the order of the denominator polynomial $A(z^{-1})$. Therefore, the TF order of system example above is unity since $A(z^{-1}) = 1 + a_1 z^{-1}$. An n-order TF with δ-sample delay can be written as follows,

$$y_k = \frac{b_\delta z^{-\delta} + b_{\delta+1} z^{-(\delta+1)} + \ldots + b_m z^{-m}}{1 + a_1 z^{-1} + \ldots a_n z^{-n}} u_k \qquad \text{for which } \delta \le m \tag{3.7}$$

Considering the general transfer function defined in equation (3.5), any pure time delay of $\delta > 0$ samples can be accounted for by setting the δ leading parameters of the polynomial $B(z^{-1})$ to zero, i.e. $b_o \ldots b_{\delta-1} = 0$. Note that for control system design purposes, it is required to have at least 1 sample delay, i.e. the b_o parameter in equation (3.5) should set to zero.

3.1.3 Controllability

It is important to examine the TF against controllability and check if it is controllable. The *minimum* requirements for controllability of the linear discrete-time TF model can be given as: Given a Single-Input Single-Output SISO discrete-time system described by TF model, equations (3.5) should satisfy the following two conditions:

1. The polynomials $A(z^{-1})$ and $B(z^{-1})$ are coprime, and
2. $\sum_{j=1}^{m} b_j \neq 0$.

3.1.4 Statistical estimation of discrete time TF models

Consider, a 1st order TF model with unity sample delay

$$\hat{y}_k = \frac{b z^{-1}}{1 + a z^{-1}} u_k \tag{3.8}$$

for which $\hat{y}_k$ is the theoretical or model response. In order to estimate the model parameters a and b, a measurement for the output y_k should be carried. The most obvious way of estimating the parameters a and b in equation (3.9) is shown in the next figure. Here, a cost function J is defined as the sum of the squares of the *response error*, e_k, i.e.

$$J = \sum_{k=1}^{k=N} e_k^2 \tag{3.9}$$

where $e_k = y_k - \hat{y}_k$. The model parameters a and b are then adjusted in some manner such as *"Least squares"* to minimize the cost function J. There are several approaches to solve this problem, two of them will be discussed in this chapter.

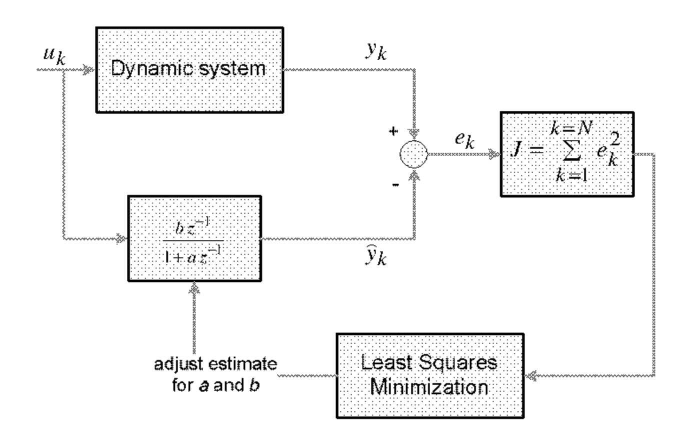


Figure 3.4 The general approach for estimating TF model parameters

1- The response error method

The main problem with this method is the fact that the error e_k is a nonlinear function of the parameter estimates. As a result, it is not possible to obtain an analytical expression for the LS estimates which minimize the cost function J and resort must be made to *numerical optimization* methods. If the value of J is plotted against the parameter estimates a and b, as indicated in the next figure, then it can be considered as a surface, the minimum of which defines the *best* estimates in a LS sense. There are many general "search procedure" for achieving this, such as simple "steepest descent" method to more complex procedures such as the *fminsearch* function available in MATLAB®.

It should be noted that, all numerical optimization methods need to be supplied with starting values for the parameter estimates and, since there may be *multiple minima* (i.e. more than one location where the gradient is zero), the search procedure may lead to *local minimum* rather than desired *global minimum* solution. Recent developments, such as "Genetic Algorithms" and "Simulated Annealing", try to overcome this potential problem.

2- The equation error method

Here, the error function is based on the model in its equation, rather than TF form. Consider equation (3.9) for which $J=\sum_{k=1}^{k=N} e_k^2=\sum_{k=1}^{k=N}(y_k-\hat{y}_k)^2$. From equation (3.8), it is known that $\hat{y}_k=-a\,\hat{y}_{k-1}+b\,u_{k-1}$. By substituting, it is now possible to find the parameter estimates that minimize the cost function J analytically since

$$J=\sum_{k=1}^{k=N}(y_k+a\,\hat{y}_{k-1}-b u_{k-1}) \tag{3.10}$$

and the minimum can be determined by calculus;

$$\begin{aligned}\frac{\partial J}{\partial a}&=2\sum_{k=1}^{k=N}(y_k+a\,\hat{y}_{k-1}-b\,u_{k-1})(y_{k-1})=0\\ \frac{\partial J}{\partial b}&=2\sum_{k=1}^{k=N}(y_k+a\,\hat{y}_{k-1}-b\,u_{k-1})(-u_{k-1})=0\end{aligned} \tag{3.11}$$

These are the conditions for the gradient of J with respect to both parameter estimated to be simultaneously equal to zero, so locating the absolute minimum J. Clearly, the conditions at (3.11) yield two simultaneous equations in the two unknown parameter estimates. This can be written in matrix form as

$$\begin{bmatrix}\sum y_{k-1}^2 & -\sum y_{k-1}u_{k-1}\\ -\sum y_{k-1}u_{k-1} & \sum u^2{}_{k-1}\end{bmatrix}\begin{bmatrix}a\\ b\end{bmatrix}=\begin{bmatrix}-\sum y_k\,y_{k-1}\\ \sum y_k\,u_{k-1}\end{bmatrix} \tag{3.12}$$

Model Evaluation

There are various statistical criteria are used for evaluating the estimated model. One particular useful criterion is called *coefficient of determination* R_T^2 which based on the model error (model residuals) e_k, this criterion can be written as

$$R_T^2 = 1 - \frac{\text{cov}(e_k)}{\text{cov}(y_k)} \tag{3.13}$$

given that

$$\text{cov}(e_k) = \frac{1}{N}\sum_{k=1}^{N}(e_k - \overline{e}_k)^2$$

$$\text{cov}(y_k) = \frac{1}{N}\sum_{k=1}^{N}(y_k - \overline{y}_k)^2$$

where $e_k = y_k - \hat{y}_k$, $\overline{e} = mean(e_k)$, and $\overline{y}_k = mean(y_k)$; given that y_k is the actual measured output and $\hat{y}_k$ is the theoretical response coming out from the model. In general R_T^2 is used as a measure of how well the model fits or explains the data: the closer to unity, the better the fit.

3.1.5 Stability and Unit Circle for Discrete-Time TF model

A linear dynamic system is said to be *stable* if its output is bounded for any bounded input. For example, in the case of continuous-time, system is stable if all *poles* (roots of the characteristic equation) lie in the left hand side of the complex *s*-plane[2], where *s* is the Laplace transform operator. The equivalent condition for a discrete-time system is based on the z^{-1} operator which states that: *for stability, the poles must have a magnitude less than unity.*

In terms of the control system, the relation between y_k and u_k defines the stability and transient dynamics (e.g. speed of response). Consider the following TF that relates both y_k and u_k,

[2] This implies that the poles must have a negative real part in order to insure decaying of the output signal to a certain definite value.

$$y_k = \frac{N(z^{-1})}{D(z^{-1})} u_k \tag{3.14}$$

for which $N(z^{-1})$ and $D(z^{-1})$ are polynomials represented in terms of the backward shift operator z^{-1}. For stability of system (3.14), the poles must have a magnitude less than unity, i.e.

$$|p_i| < 1 \qquad \forall \;\; i = 1, 2, \ldots, n \tag{3.15}$$

where n is the order of the TF model (3.15) and p_i are the poles, i.e. the roots of $D(z^{-1}) = 0$. In other words,

$$D(z^{-1}) = (1 - p_1 z^{-1})(1 - p_2 z^{-1}) \cdots (1 - p_n z^{-1}) = 0 \tag{3.16}$$

However, the polynomial $D(z^{-1})$ is written in terms of the backward shift operator z^{-1}, it is more convenient to define it in terms of z. Fortunately, it is straightforward to convert from z^{-1} to z domain by simply multiply the characteristic equation by z^n.

If the poles of the characteristic equation $D(z^{-1}) = 0$ are nearby unity, the response will be too slow. However, if the poles are greater than unity, the response will never reaches a definite value and therefore the system is said to be *unstable*, see Figure (3.5).

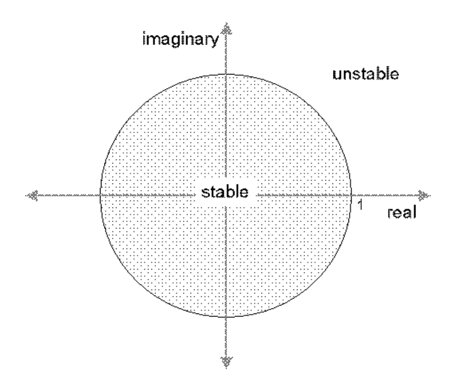


Figure 3.5 Unit circle and stability for discrete time system

3.2 Discrete Time Model Based Control

The notation and terminology for an automatic feedback control system, as employed in the present work is summarized in Figure (3.6), where y_k represents the discrete-time value of the measured output variable at the k^{th} sample. Remember that in an open loop control system, the output is not measured and the controller cannot respond to disturbances or unexpected variations in the output. By contrast, in Figure (3.6), the controller operates on the error signal between the system output y_k and the command or reference input r_k, where the latter is sometimes called the set-point or the desired level. Also shown in Figure (3.6) is a feedback of the implemented control input variable u_k and the significance of this latter feedback loop will be discussed later.

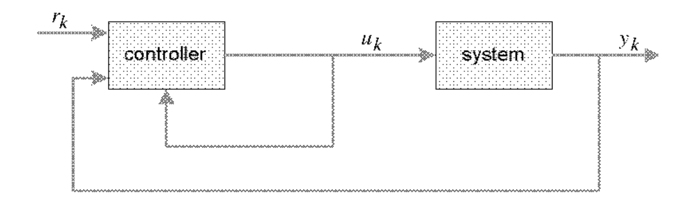


Figure 3.6 General layout of automatic feedback control system

Generally, the objectives of the control system are as follows:

(i) The closed loop system should be stable;

(ii) The output should 'track' the command input (zero steady state error);

(iii) The transient behaviour should be satisfactory in some sense (speed of response, overshoot, oscillations, etc.); and

(iv) The control system should reject unwanted disturbances.

A flow chart describing one particular iterative approach to control system design is illustrated in Figure (3.7), which starts with the identification of the discrete-time model

which based on statistical methods to identify the model on the basis of data collected during laboratory or field studies, i.e. a data-based approach[3]. This model is used as a surrogate for the real system, in order to evaluate the performance of the controller before moving to the implementation stage.

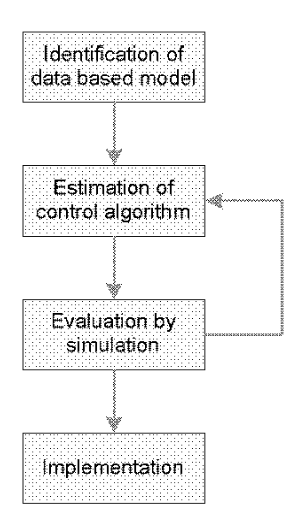


Figure 3.7 Control system design process

3.2.1 Basics of discrete-time model-based control

Once the discrete-time TF is identified, estimation of control algorithm starts to take its place, see Figure (3.7), in order to control the system. This control algorithm should satisfy the four major objectives described above.

The most common building blocks for discrete-time control systems are:

1. Proportional controller.
2. Integral controller.
3. Proportional-Integral PI controller.

[3] The approach of data based modelling has been discussed in the previous section.

The later is the standard approach for the control of 1st order systems. The three control algorithms are listed below with their equivalent block diagram.

1- Proportional (P) controller

The control equation of this type of control, based on discrete (incremental) form, is

$$u_k = k_p e_k \tag{3.17}$$

This can represented in a block diagram as shown in Figure (3.8). Here, y_k represents the measured output variable at the k^{th} sample, u_k the control input, r_k the set or reference point, and $e_k = r_k - y_k$ is the error between the set point and the measured output y_k. The above control equation is written in an incremental form which allows straightforward implementation in practice, using a standard computer language such as "C" or in a software package like the real-time toolbox of Matlab®. It is important to note that the only *state* used here for the control law is the error e_k.

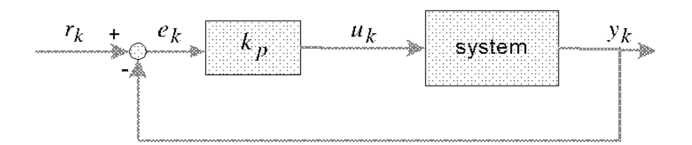


Figure 3.8 Block diagram representation of P-controller

2- Integral (I) controller

The incremental form in this controller is:

$$u_k = u_{k-1} + k_I e_k \tag{3.18}$$

This can be written in the form

$$u_k = \frac{k_I}{1 - z^{-1}} e_k \tag{3.19}$$

Therefore, the block diagram can take the form in Fig.(4).

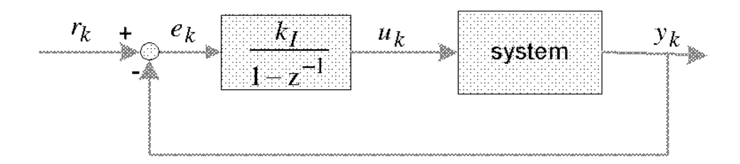


Figure 3.9 Block diagram representation of I-controller

The only *state* existed is the integral of error $z_k = \frac{1}{1-z^{-1}} e_k$. This helps to improve the speed of tracking performance.

3- Proportion-Integral (PI) controller

By combining the above proportional and integral control action, the incremental form of PI control can be written as follows,

$$u_k = u_{k-1} + k_I e_k - k_p [y_k - y_{k-1}] \tag{3.20}$$

which can be rearranged as

$$u_k (1 - z^{-1}) = k_I e_k - k_p y_k (1 - z^{-1}) \tag{3.21}$$

therefore it can written as

$$u_k = \frac{k_I}{1-z^{-1}} e_k - k_p y_k \tag{3.22}$$

Here, there are two *states* are used in the control law, they are the integral of error state $z_k = 1/(1 - z^{-1}) e_k$ and the output state y_k, therefore, the block diagram takes the form depicted in Figure (3.10). However, the control gains k_p and k_I are tuned on the basis of experience. Alternatively, a number of control techniques are available to estimate their values such as Bode and Nyquist plots, Root Locus method and pole assignment technique.

The advantage of the latter approach is that it utilizes the previously identified TF model to directly estimate the appropriate control gains. This approach will be discussed in the following example.

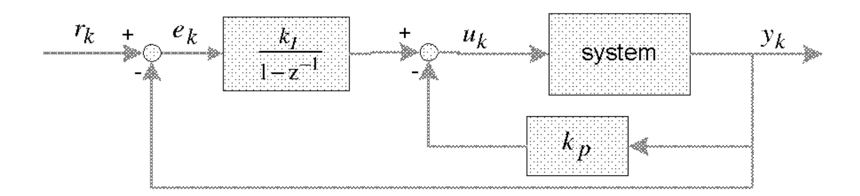


Figure 3.10 Block diagram representation of Proportional-Integral (PI) controller

Example

Consider the reservoir system, equation (3.4), for which $a = -0.8$ and $b = 0.2$. It is required now to design the three different controllers for the same system using P-, I- and PI-controllers.

Proportional controller

The block diagram will be as shown in Figure (3.8). This can be reduced as

$$r_k \rightarrow \boxed{\frac{k_p\, b\, z^{-1}}{1+(a+k_p\, b)z^{-1}}} \rightarrow y_k$$

Figure 3.11 The reduced block diagram of P-controller

For system stability, the poles *"roots"* of the characteristic equation $D(z^{-1}) = 1+(a+k_p\, b)\, z^{-1} = 0$ should be less than unity, as discussed earlier in this chapter. By multiplying both sides of the characteristic equation by z, then

$$z+(a+k_p\, b) = 0 \qquad (3.23)$$

therefore, the pole $p_1 = -a - k_p b$ should be inside the unit circle, i.e.

$$-a - k_p b < 1 \qquad k_p > \frac{-a-1}{b}$$

$$-a - k_p b > -1 \qquad k_p < \frac{-a+1}{b}$$

The fastest possible *"deadbeat"* response should be attained when the pole $p_1 = -a - k_p b = 0$, therefore

$$k_p = \frac{-a}{b} \tag{3.24}$$

Given that $a = -0.8$ and $b = 0.2$; therefore in order to achieve stable response, the proportional gain should be in the range $9 < k_p < -1$. However, in order to achieve the deadbeat response the gain should be $k_p = 4$. Figure (3.13) shows the SIMULINK® diagram with two different responses at unity set point. Also, its equivalent block diagram is depicted in Figure (3.12).

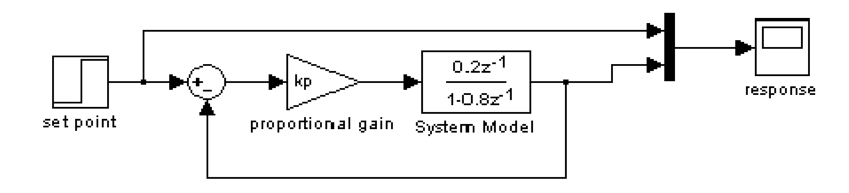


Figure 3.12 The SIMULINK® block diagram for P-controller for example (1)

However Figure (3.13) shows stable response, the steady state response y_∞ does not reach the required level r_∞. To understand this behaviour, consider the closed-loop TF in Figure (3.11). Since the steady state response can be attained by setting $z^{-1} = 1$, therefore

$$y_\infty = \frac{k_p b}{1 + a + k_p b} r_\infty \tag{3.25}$$

It is obvious that $y_\infty \neq r_\infty$ because $a \neq -1$.

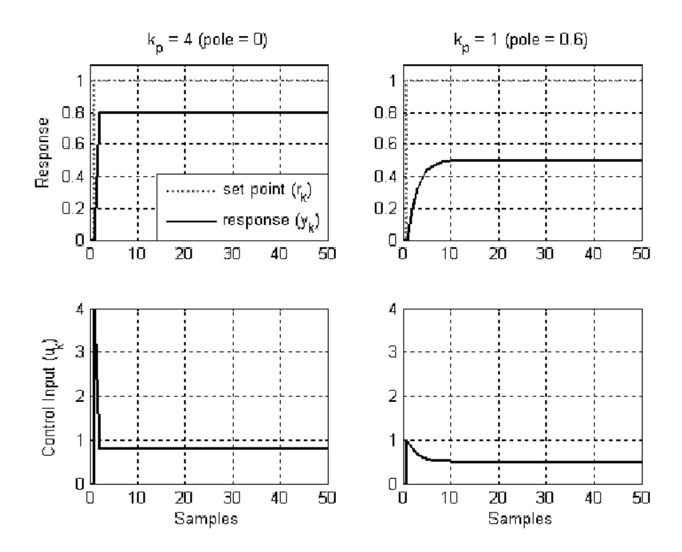


Figure 3.13 Response for system example (1) using P- controller at two different poles

Integral controller

In this case, the block diagram will be as shown in Figure (3.9) which can be reduced as

$$r_k \rightarrow \boxed{\frac{k_I\, b\, z^{-1}}{1+(a-1+k_I\, b)z^{-1}-a\, z^{-2}}} \rightarrow y_k$$

Figure 3.14 The reduced block diagram of I-controller

Unlike P-controller, the closed loop function shown in the above figure confirm that the steady state response y_∞ always equals to the set point r_∞ whatever the value of the integral gain k_I , see Figure (3.26). This can be verified easily by setting $z^{-1}=1$. Also, the characteristic equation is

$$D(z^{-1})=1+(a-1+k_I\, b)z^{-1}-a\, z^{-2}=0 \tag{3.26}$$

For simplicity, this can be written as

$$\begin{aligned} z^2+(a-1+k_I\, b)z-a=0 \\ (z+p_1)(z+p_2)=0 \end{aligned} \tag{3.27}$$

It is obvious that the dead beat response ($p_1 = p_2 = 0$) cannot be attained unless the free term a equals to zero. Also, if the free term is greater than unity, it is not possible to achieve steady state response since the poles or one of the poles in this case will be outside the unit circle. Moreover, in the case of free term less than unity, the selection of poles are not easy since the poles selected should be $p_1 p_2 = a$. In order to understand this point, substitute $a = -0.8$ and $b = 0.2$ in the characteristic equation above, then

$$z^2 + (k_I b - 1.8) z + 0.8 = 0 \qquad (3.28)$$

The poles may be as $p_1 = p_2 = 0.8944$ such that $p_1 \times p_2 = 0.8$. In this case the integral gain will be $k_I = 0.0557$. The SIMULINK® diagram and the response are depicted in Figures (3.15 and 3.16) respectively for which the response matches the set point r_k.

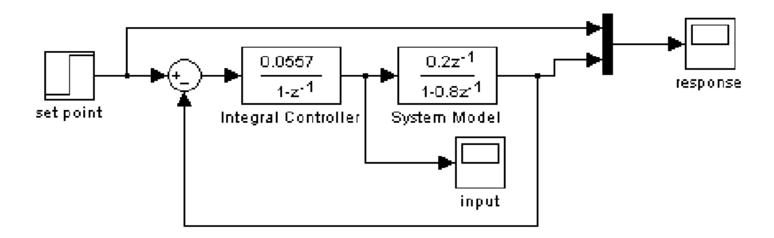


Figure 3.15 The SIMULINK® block diagram for I-controller for example (1)

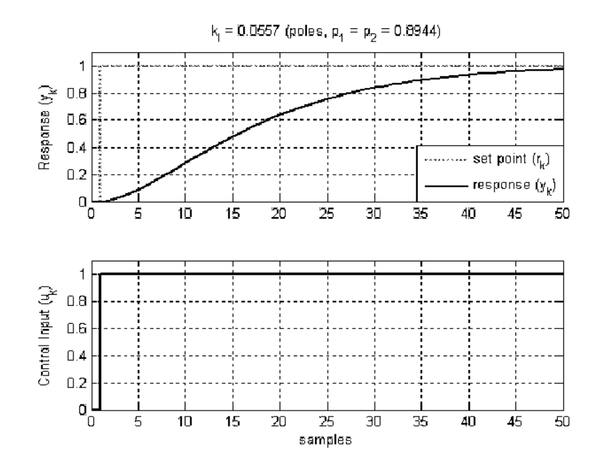


Figure 3.16 Response for system example (1) using I-controller

Proportional-Integral controller

Here, the block diagram for PI-controller takes the form in Figure (3.10) which can be reduced as

$r_k \rightarrow$ $\dfrac{k_I\, b\, z^{-1}}{1+(a-1+k_p\, b+k_I\, b)z^{-1}-(a+k_p\, b)\, z^{-2}}$ $\rightarrow y_k$

Figure 3.17 The reduced block diagram of PI-controller

Similar to I-control, the closed loop function shown in Figure (3.17) confirms that the steady state response y_∞ always equals to the set point r_∞ whatever the value of both proportional and integral gains (k_p and k_I). This can be verified easily by setting $z^{-1}=1$. Also, the characteristic equation is

$$D(z^{-1}) = 1+(a-1+k_p\, b+k_I\, b)z^{-1}-(a+k_p\, b)\, z^{-2} = 0 \tag{3.29}$$

which can be simplified as

$$\begin{aligned} z^2+(a-1+kp\, b+k_I\, b)\, z-(a+k_p\, b) &= 0 \\ (z+p_1)(z+p_2) &= 0 \end{aligned} \tag{3.30}$$

Unlike I-controller, the dead beat response ($p_1 = p_2 = 0$) can be attained, this is because the free term $(a+k_p\, b)$ can be adjusted to be equal to zero due to the existence of proportional gain k_p. Also, any value of poles (inside the unit circle) can be selected due to the flexibility existed in characteristic equation, therefore, the required speed of response can be achieved easily. In order to understand this point, substitute $a=-0.8$ and $b=0.2$ in the characteristic equation above, then

$$z^2+0.2(k_p+k_I-9)\, z+(0.8-0.2k_p) = 0 \tag{3.31}$$

The next table shows some selected poles and their corresponding gains

poles required	corresponding characteristic equation	corresponding gains
$p_1 = p_2 = 0$	$z^2 + 0z + 0 = 0$	$k_p = 4$ $k_I = 5$
$p_1 = p_2 = 0.5$	$z^2 - z + 0.25 = 0$	$k_p = 2.75$ $k_I = 1.25$
$p_1 = -0.1,\ p_2 = 0.6$	$z^2 - 0.5z + 0.06 = 0$	$k_p = 3.7$ $k_I = 5.3$

The SIMULINK® block diagram and the responses of the three selected gains are shown in Figures (3.18 and 3.19).

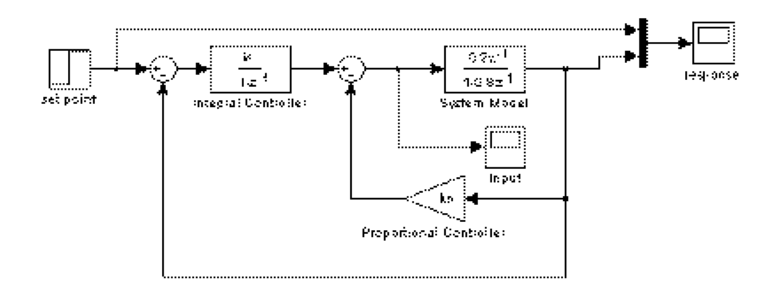


Figure 3.18 The SIMULINK® block diagram for PI-controller for example (1)

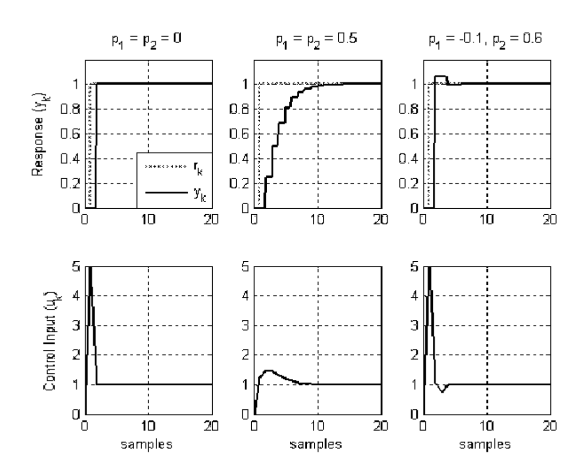


Figure 3.19 Response of example (1) using proportional integral controller

A quick look at the above three basic control algorithm, the following points can be observed:

1- The response y_k of the proportional controller sometimes cannot track the command level or set point r_k. However, it is possible to get the deadbeat response.

2- The selection of poles for integral controller is difficult, also the deadbeat response cannot be achieved. Finally, the response of integral controller is relatively slow.

3- The above two points lead to the following fact: *The order of the characteristic equation should equal the number of gains in order to achieve fast response and flexible poles choice.*

4- Proportional integral controller shows the fastest and the most flexible design among the three controllers. This is because the number of gains equals the order of the characteristic equation.

3.2.2 PIP control

Proportional Integral Plus (PIP) control is a modern conventional control which is based on True Digital Control (TDC). PIP can be interpreted as a logical extension of conventional PI controller with additional dynamic feedback and input compensators introduced automatically when the process has second order or higher dynamics, or pure time delays greater than unity.

1- NMSS representation

Once the discrete-time TF model is identified and estimated, it is easy to show that this model can be represented in the following NMSS form

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{F}\, x_{k-1} + \boldsymbol{g}\, u_{k-1} + \boldsymbol{d}\, r_k \\ y_k &= \boldsymbol{h}\, \boldsymbol{x}_k \end{aligned} \tag{3.32}$$

for which $\boldsymbol{x}_k = [y_k \ \cdots \ y_{k-n+1} \ u_{k-1} \ \cdots \ u_{k-m+1} \ z_k]^T$ is the $n+m$ dimensional state vector, consisting of the present and past values of the output variable y_k and the past values of the input u_k. With this special NMSS representation, all the states are directly measurable. Moreover, the NMSS description includes the following integral-of-error state z_k to ensure tracking performance,

$$z_k = z_{k-1} + (r_k - y_k) \tag{3.33}$$

for which r_k is the reference or command input. Finally, the state transient matrix $\boldsymbol{F}$, input vector $\boldsymbol{g}$, command input vector $\boldsymbol{d}$ and output vector $\boldsymbol{h}$ are defined as follows,

$$\boldsymbol{F} = \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_{n-1} & -a_n & b_2 & b_3 & \cdots & b_{m-1} & b_m & 0 \\ 1 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ a_1 & a_2 & \cdots & a_{n-1} & a_n & -b_2 & -b_3 & \cdots & -b_{m-1} & -b_m & 1 \end{bmatrix}$$

$$\begin{aligned} \boldsymbol{g} &= [b_1 \ \ 0 \ \ \cdots \ \ 0 \ \ 1 \ \ 0 \ \ 0 \ \ \cdots \ \ 0 \ \ 0 \ \ -b_1]^T \\ \boldsymbol{d} &= [0 \ \ 0 \ \ \cdots \ \ 0 \ \ 0 \ \ 0 \ \ 0 \ \ \cdots \ \ 0 \ \ 0 \ \ 1]^T \\ \boldsymbol{h} &= [1 \ \ 0 \ \ \cdots \ \ 0 \ \ 0 \ \ 0 \ \ 0 \ \ \cdots \ \ 0 \ \ 0 \ \ 0] \end{aligned} \tag{3.34}$$

2- PIP Control Design

The state variable feedback (SVF) control law associated with the NMSS model represented in equation (3.33) then takes the form,

$$u_k = -\boldsymbol{k}\,\boldsymbol{x}_k \qquad (3.35)$$

where $\boldsymbol{k}$ is the $n+m$ dimensional SVF control gain vector,

$$\boldsymbol{k} = [f_o \;\; f_1 \;\; \cdots \;\; f_{n-2} \;\; f_{n-1} \;\; g_1 \;\; g_2 \;\; \cdots \;\; g_{m-2} \;\; g_{m-1} \;\; -k_I]^T \qquad (3.36)$$

This SVF control gain vector not only contains proportional and integral gains, like a PI controller, but also includes higher order input and output feedback compensators which enhance the performance of the controller as depicted in Figure (3.20). As shown in this figure, PIP control can be considered as an extension of PI control, for which the PI action is enhanced by the higher order input and output feedback compensators $G_1(z^{-1})$ and $F_1(z^{-1})$ respectively, where

$$G_1(z^{-1}) = g_1 z^{-1} + g_2 z^{-2} + \cdots g_{m-1} z^{-(m-1)}$$
$$F_1(z^{-1}) = f_1 z^{-1} + f_2 z^{-2} + \cdots + f_{n-1} z^{-(n-1)} \qquad (3.37)$$

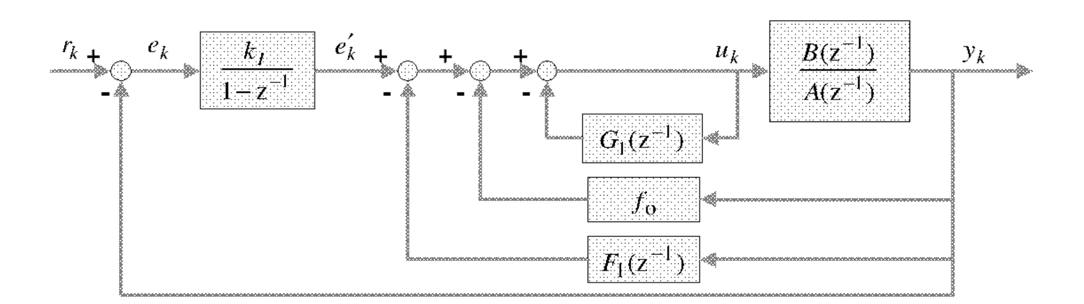


Figure 3.20 The PIP control system implemented in standard feedback form.

Since all the state variables in $\boldsymbol{x}_k$ are readily stored in the digital computer, the PIP controller can be implemented easily. Moreover, the inherent SVF formulation allows for the exploitation of any SVF procedure such as arbitrary pole assignment of the closed loop dynamics.

In the case of SVF pole assignment, it is required to get the closed-loop control system in transfer function form, and this could be obtained directly by reducing the block diagram shown in Figure (3.20) as follows,

$$y_k = \frac{k_I\, B(z^{-1})}{\Delta\left[G(z^{-1})A(z^{-1}) + F(z^{-1})B(z^{-1})\right] + k_I B(z^{-1})} r_k \tag{3.38}$$

for which $\Delta = 1 - z^{-1}$ is the difference operator, $F(z^{-1}) = f_o + F_1(z^{-1})$ is the output feedback compensator and $G(z^{-1}) = 1 + G_1(z^{-1})$ is the input compensator. Then a simple polynomial algebra of the following characteristic equation

$$\begin{aligned} &\Delta\left[G(z^{-1})A(z^{-1}) + F(z^{-1})B(z^{-1})\right] + k_I B(z^{-1}) \\ &\qquad = (1 - p_1 z^{-1})(1 - p_2 z^{-1}) \cdots (1 - p_{n+m} z^{-1}) \end{aligned} \tag{3.39}$$

could be utilized to find the SVF gains at pre-determined positions of the poles p_1, p_2, ..., p_{n+m} in the complex z-plane.

3.3 Control Performance

There are several tests to evaluate the performance of controllers, here three tests will be considered. These tests are:

1- Robustness test (Mont Carlo analysis)

Since the control gains are designed according to the estimated model, and this model has been estimated according to lab or field test, therefore there is a possibility that the

model does not match with actual dynamic behaviour of the system. This will lead to unsatisfactory behaviour (such as overshooting, oscillation about the set point, ..., etc) for the controller when implemented to the actual system due to this model mismatch. Therefore, it is very important to measure the percentage of model parameter variation when using the same selected gains. This percentage gives an indication about how much does the controller *robust* from the point of view of model mismatch.

2- Input disturbance rejection

During actual implementation of the controller, the system could be subjected to unwanted *input disturbance*, therefore it is very important to simulate this disturbance to ensure that the design is safe against any kind of these disturbances.

3- Output disturbance rejection

Similarly, the actual system could be subjected to any kind of disturbance that may affect the output. For example the control system used for Air Condition (AC) systems may be subjected to *output disturbance* if unexpected heat wave hit the system. Therefore, it is very important to simulate this disturbance to make sure that the designed controller can overcome and damp these disturbances if existed.

Table of Contents

List of Figures

CHAPTER 4

HEATING SYSTEM CONTROL

The gas burner in Cellopack Company, see Figure (2.2), is based on natural gas for which the flow rate of the natural gas is controlled by means of mechanical gate. This gate is motorised by DC motor and the opening of the gate is measured and feedback by means of digital encoder, as discussed in Section (2.1). Relays are existing to actuate the DC motor in both direction to close or open the gas gate. The hot air is passing through isolated ducts to the target machine to dry the liquid-based ink. At the end of the duct, just before the printing machine, there is a PT100 which feedbacks the current temperature to the controller, see Figure (4.1). The role of the controller is to determine the proper opening of the gate such that the temperature measured is around the required temperature.

The target control system is automatically control the opening of the natural gas gate in order to set the hot air temperature to the required level according to the worker selection by means of the set point selector. The performance of this stage should be

$\pm 5\ ^{o}C$ with settling time no more than 30 minutes. This should be done regardless the state of the fresh air blower (ON or OFF). Also, half the amount of the recycled hot air is permanently recycled to the gas burner. Here, the system is SISO (single-input, single-output). The input is the position of the natural gas gate, and the output is the temperature of the hot air, see Figure (4.1).

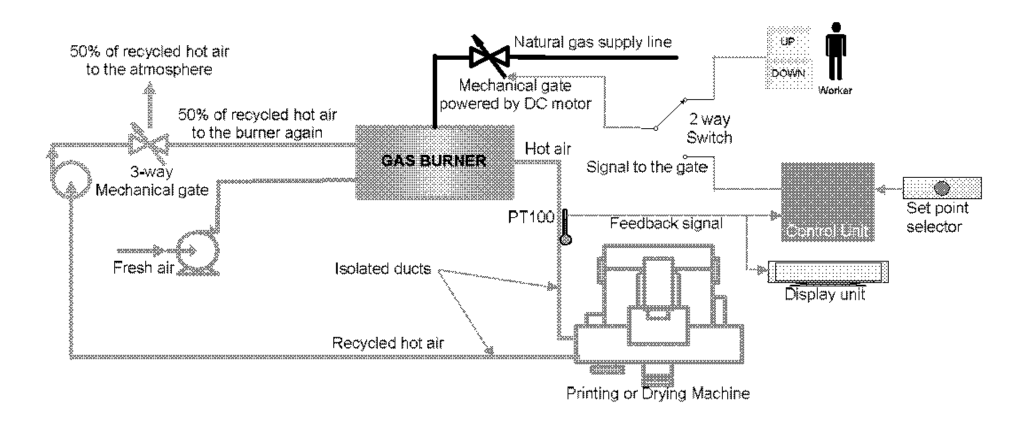


Figure 4.1 The developed control system

Originally the system was working manually[1], see Figure (2.5). By contrast, the present work develops a complete control system and evaluates several algorithms for controlling the temperature of the hot air. It is convenient to note that the implementation of PIP control (introduced in Chapter 3) represents one of the first applications of such methods to a real system in Egypt.

4.1 Dynamics for the Hot Air System

Since the system is a heat transfer system, then it has relatively slow response for which it was found that 4 sec sample delay is good enough to evaluate the dynamic behaviour of the system.

[1] This is because the damage of the PLC controller of the gas burner.

4.1.1 Linear model identification

In the first instance, linear TF models are identified and estimated using response error method by means *fminsearch* function existing in Matlab®. The coefficient of determination, R_T^2 is used to evaluate the estimated model. Step input functions are used to excite the system, with data collected for different magnitudes. The magnitudes are: $u = 1$, 2, 3, 4, 5, 6, 7, 8, 10 and 12. Also, two data files were collected using random signal which ranges from $u = 0 \rightarrow 13$. In order to collect these data, a program using LabView® has been created as shown in Figure (4.2).

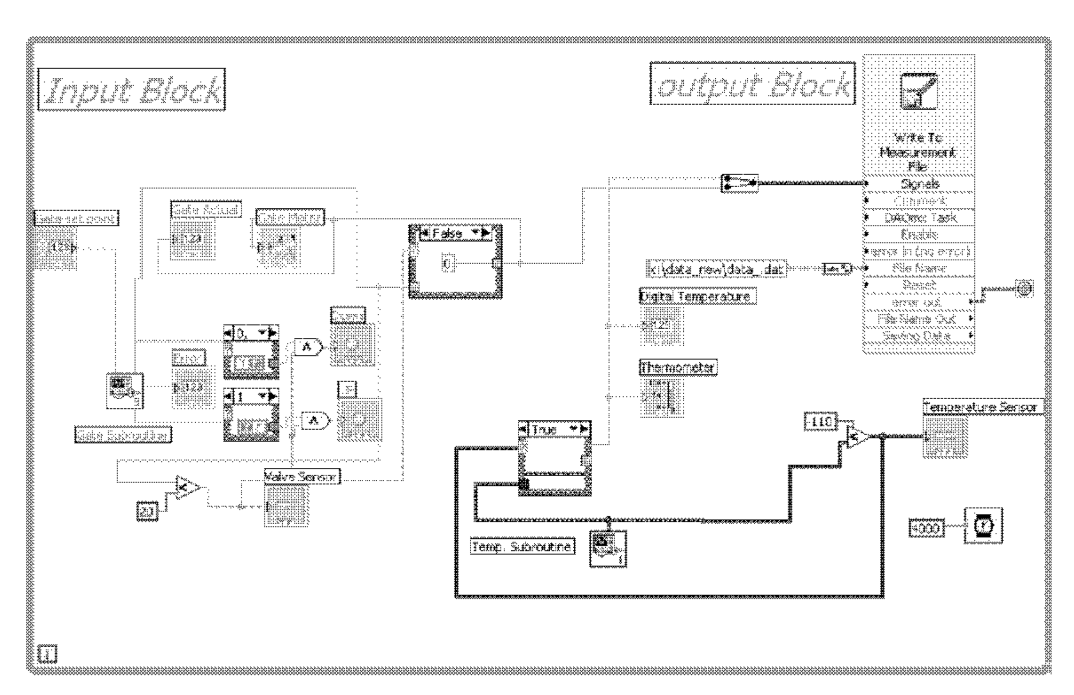


Figure 4.2 LabView® program used for collecting data

There are several models which can be well-define the dynamic behaviour of the system. In order to select the model which best define the system without over-parameterisation, the coefficient of determination, R_T^2, is utilised.

Model (1-1-1)

Here the model has the following form,

$$y_k = \frac{b_1 z^{-1}}{1 + a_1 z^{-1}} u_k \tag{4.1}$$

Gate	01	02	03	04	05	06	07	08	10	12	PRBS	PRBS1
R_T^2	0.6707	0.9673	0.9778	0.9904	0.9929	0.9923	0.9915	0.8412	0.8827	0.9717	0.8912	0.7453
a_1	-1.0008	-1.0030	-1.0045	-0.9536	-0.9884	-0.9490	-0.9934	-0.9952	-0.9921	-0.9732	-0.9592	-0.9727
b_1	-0.0189	-0.0098	-0.0276	0.7923	0.1972	0.7721	0.1144	0.1153	0.4794	0.5842	0.7288	0.4170

Model (1-1-2)

Here the model has the following form,

$$y_k = \frac{b_2 z^{-2}}{1 + a_1 z^{-1}} u_k \tag{4.2}$$

Gate	01	02	03	04	05	06	07	08	10	12	PRBS	PRBS1
R_T^2	0.6911	0.9738	0.9834	0.9895	0.9938	0.9927	0.9909	0.9965	0.9697	0.9805	0.8838	0.7266
a_1	-1.0008	-1.0036	-1.0051	-0.9526	-0.9882	-0.9486	-0.9934	-1.0075	-0.9945	-0.9720	-0.9570	-0.9718
b_2	-0.0219	-0.0266	-0.0382	0.8104	0.1993	0.8046	0.1128	0.1000	0.4845	0.5963	0.7681	0.4288

Model (2-1-2)

Here the model has the following form,

$$y_k = \frac{b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} u_k \tag{4.3}$$

Gate	01	02	03	04	05	06	07	08	10	12	PRBS	PRBS1
R_T^2	0.7315	0.9750	0.9841	0.9901	0.9937	0.9927	0.9913	0.9992	0.9994	0.9825	0.9460	0.8729
a_1	-1.7404	-1.1514	-1.1102	-0.1598	-0.4354	-0.2308	-0.1524	-1.6461	-1.9230	-1.4563	-1.6856	-1.2970
a_2	0.7800	0.1960	0.1678	-0.7604	-0.4756	-0.6906	-0.7649	0.6662	0.9446	0.4915	0.7122	0.3452
b_2	0.0390	0.1668	0.1772	0.5036	0.4134	0.6001	0.5092	0.1602	0.1097	0.3429	0.1997	0.3966

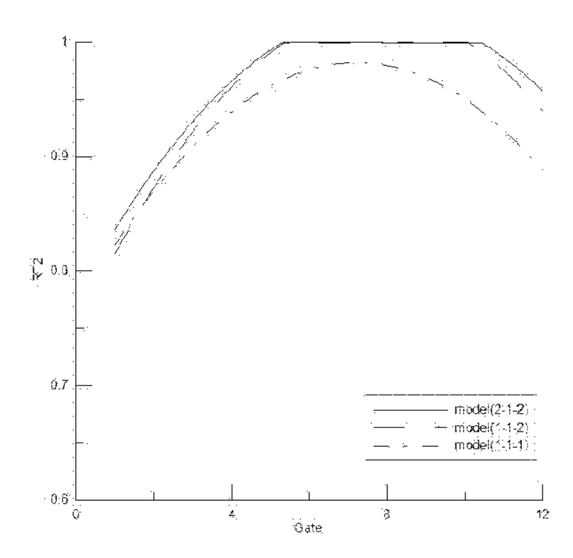


Figure 4.3 Coefficient of determination, R_T^2, for several proposed models

Figure (4.3) compares R_T^2 for the three proposed models. It is obvious that the model (2-1-2) defined in equation (4.3) shows higher R_T^2.

4.1.2 Linear model estimation

The next step is estimating the parameters of the estimated model (4.3). By means of the collected data several model parameters are achieved according to the gate opening or u. The values of these parameters are tabulated below.

u	01	02	03	04	05	06	07	08	10	12	PRBS	PRBS1
R_T^2	0.7315	0.9750	0.9841	0.9901	0.9937	0.9927	0.9913	0.9992	0.9994	0.9825	0.9460	0.8729
a_1	-1.7404	-1.1514	-1.1102	-0.1598	-0.4354	-0.2308	-0.1524	-1.6461	-1.9230	-1.4563	-1.6856	-1.2970
a_2	0.7800	0.1960	0.1678	-0.7604	-0.4756	-0.6906	-0.7649	0.6662	0.9446	0.4915	0.7122	0.3452
b_2	0.0390	0.1668	0.1772	0.5036	0.4134	0.6001	0.5092	0.1602	0.1097	0.3429	0.1997	0.3966

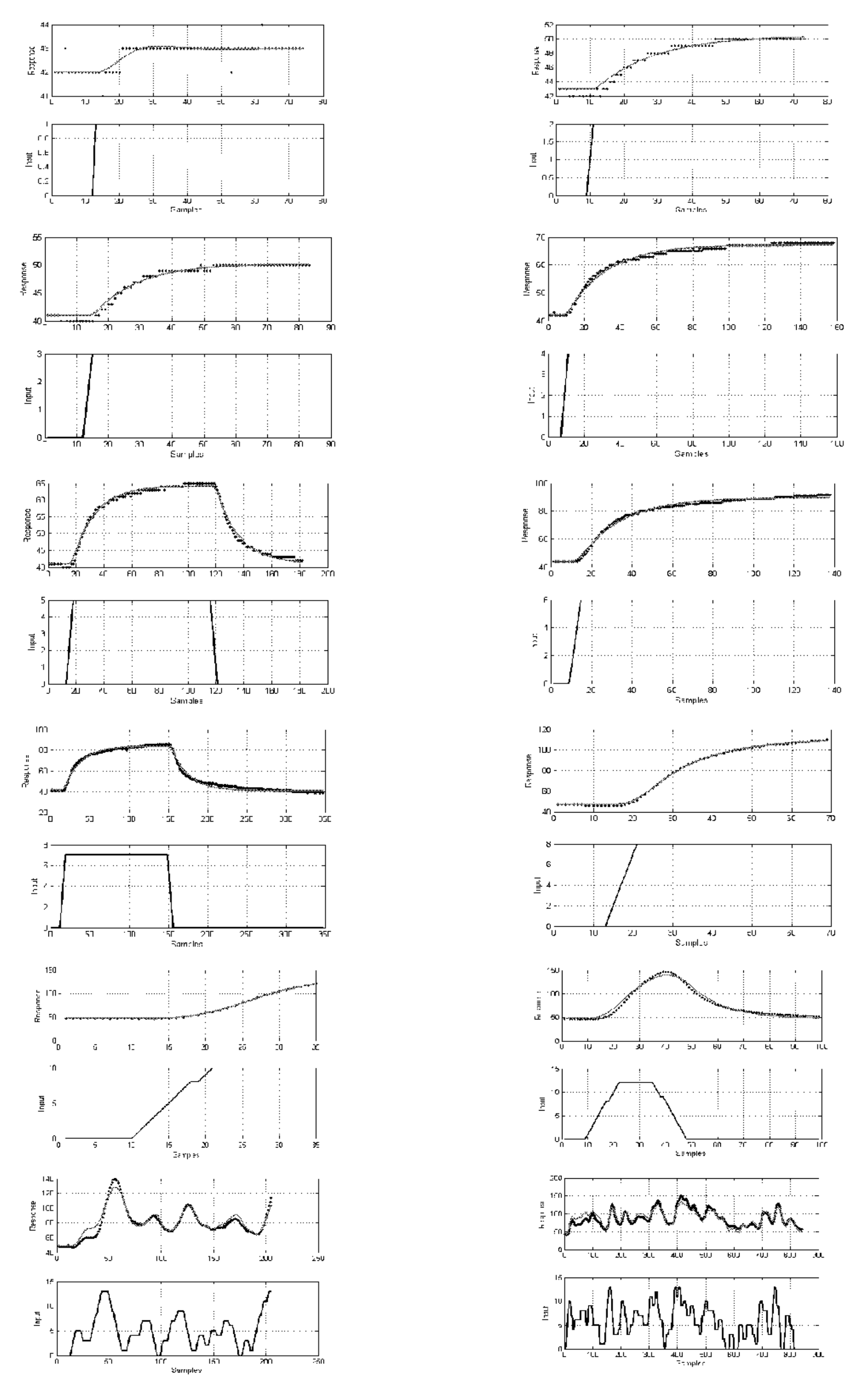


Figure 4.4 Model response for each data collected

The above table shows that the average values for a_1 and a_2 are -1 and $+0.05$ respectively. Therefore, these values will be selected and fixed and then estimating the corresponding b_2 at each gate opening. The results are depicted in the following table.

u	01	02	03	04	05	06	07	08	10	12
R_f^2	*0.7315*	*0.9750*	*0.9841*	*0.9901*	*0.9937*	*0.9927*	*0.9913*	*0.9992*	*0.9994*	*0.9825*
b_2	0.0610	0.1902	0.1648	0.3092	0.2348	0.3738	0.3041	0.4210	0.5472	0.5321

Figure (4.5) shows the change of b_2 versus the gate opening. This graph shows that the mean value which could be selected and best represent the data is $b_2 = 0.3$.

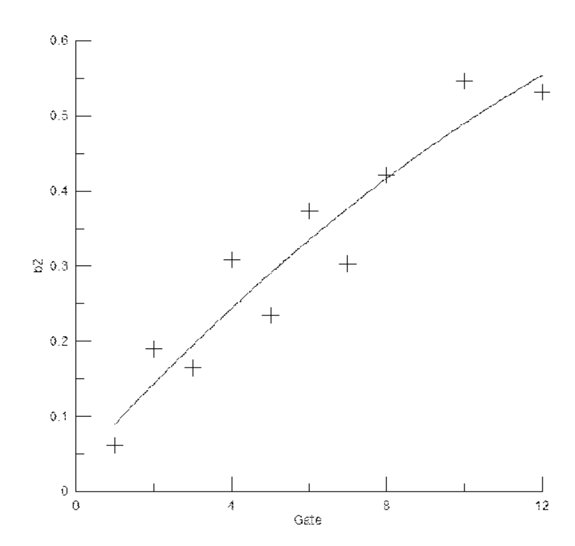


Figure 4.5 The change of b_2 versus the change of the normalized input, u, fixing the dominator parameters, $a_1 = -1$ and $a_2 = 0.05$

4.2 P-, I-, PI- and PIP Controller Design

In this subsection, P-, I-, PI- and PIP controllers are developed for heating system. Here, the control gains are determined by using pole assignment method.

4.2.1 P-Controller Design

Here, the control law will take the form of equation (3.18). The proportional gain k_p can be selected utilising the closed-loop TF depicted in Figure (3.4). By selecting poles $p_1 = p_2 = 0.5$, it is found that $k_p = 0.666$. The simulation will take the form depicted in Figure (4.6).

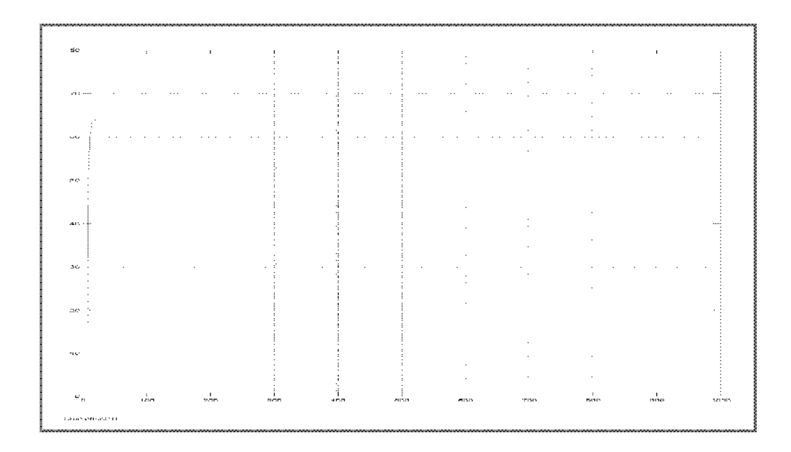


Figure 4.6 P-controller simulation using Matlab®

4.2.2 I-Controller Design

Here, the control law will take the form of equation (3.19). The integral gain k_I can be selected utilising the closed-loop TF depicted in Figure (3.14). Trial and error procedure suggest that the $k_I = 0.0027$ which corresponds to poles $p_1 = 0.9736 + 0.0122i$,

$p_2 = 0.9736 - 0.0122i$, and $p_3 = 0.0527$. The simulation will take the form depicted in Figure (4.7).

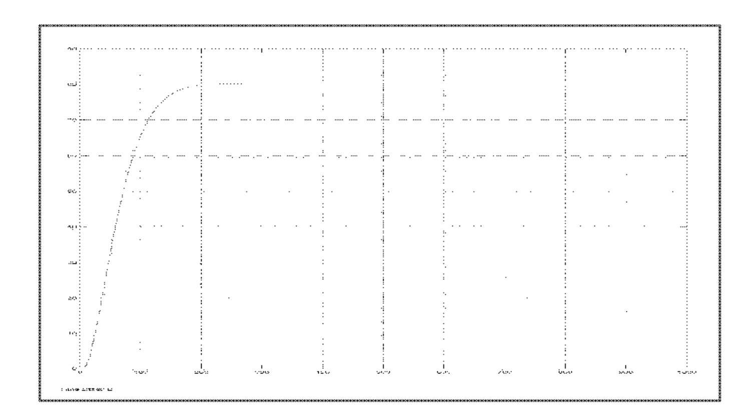


Figure 4.7 I-controller simulation using Matlab®

4.2.3 PI-Controller Design

Here, the control law will take the form of equation (3.21). The proportional and integral gains k_p and k_I can be selected utilising the closed-loop TF depicted in Figure (3.17). By selecting poles $p_1 = p_2 = p_3 = 0.667$, it is found that $k_p = 0.821$ and $k_I = 0.123$. The simulation will take the form depicted in Figure (4.8).

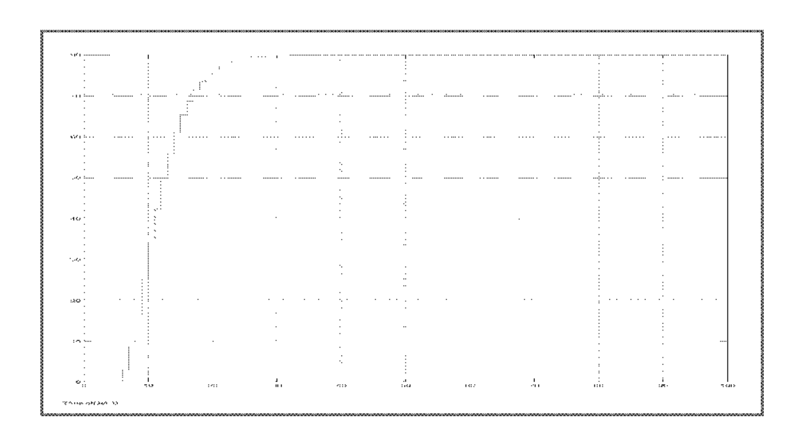


Figure 4.8 PI-controller simulation using Matlab®

4.2.4 PIP Controller Design

Here, the SVF control law will take the form of equations (3.36 and 3.37). The proportional gains k_{p_1} and k_{p_2}, input gain k_u, and integral gain k_I can be selected utilising the closed-loop TF depicted in Figure (3.39). Here, the closed loop TF will be

$$\begin{aligned} y_k &= \frac{k_I(0.3z^{-2})}{(1-z^{-1})[(1+k_{u_1}z^{-1})(1-z^{-1}+0.05z^{-2})+(k_p+k_{p1}z^{-1})(0.3z^{-2})]+k_I(0.3z^{-2})} r_k \\ &= \frac{k_I(0.3z^{-2})}{(1-z^{-1})[1-z^{-1}+0.05z^{-2}+k_{u_1}z^{-1}-k_{u_1}z^{-2}+0.05k_{u_1}z^{-3}+0.3k_p z^{-2}+0.3k_{p1}z^{-3}]+k_I(0.3z^{-2})} r_k \end{aligned} \tag{4.4}$$

Therefore, the characteristic equation will be

$$\begin{aligned} D(z^{-1}) = 1-(2-k_{u1})z^{-1}+(1.05-2k_{u1}+0.3k_p+0.3k_I)z^{-2} \\ -(0.05-1.05k_{u1}-0.3k_{p1}+0.3k_p)z^{-3}+(-0.05k_{u1}-0.3k_{p1})z^{-4} \end{aligned} \tag{4.5}$$

which can be rewritten as

$$\begin{aligned} D(z) = z^4-(2-k_{u1})z^3+(1.05-2k_{u1}+0.3k_p+0.3k_I)z^2 \\ -(0.05-1.05k_{u1}-0.3k_{p1}+0.3k_p)z+(-0.05k_{u1}-0.3k_{p1}) \end{aligned} \tag{4.6}$$

The poles should have the following form

$$\begin{aligned} (z-p_1)(z-p_2)(z-p_3)(z-p_4) = z^4-(p_1+p_2+p_3+p_4)z^3 \\ +(p_1p_2+p_1p_3+p_2p_3+p_1p_4+p_2p_4+p_3p_4)z^2 \\ -(p_1p_2p_3+p_1p_2p_4+p_1p_3p_4+p_2p_3p_4)z+p_1p_2p_3p_4 \end{aligned} \tag{4.7}$$

By comparing equation (4.6) with equation (4.7), then

$$\begin{aligned} & p_1+p_2+p_3+p_4 = 2-k_{u1} \\ & p_1p_2+p_1p_3+p_2p_3+p_1p_4+p_2p_4+p_3p_4 = 1.05-2k_{u1}+0.k_p+0.3k_I \\ & p_1p_2p_3+p_1p_2p_4+p_1p_3p_4+p_2p_3p_4 = 0.05-1.05k_{u1}-0.3k_{p1}+0.3k_p \\ & p_1p_2p_3p_4 = -0.05k_{u1}-0.3k_{p1} \end{aligned} \tag{4.8}$$

By solving system of equation depicted in equation (4.8) simultaneously, the gains corresponding to the selected poles can be calculated. The gains corresponding to poles $p_1 = p_2 = p_3 = p_4 = 0.4$ can be found as

$$
\begin{aligned}
k_{u1} &= 0.4 \\
k_{p1} &= -0.152 \\
k_{p} &= 1.93 \\
k_{I} &= 0.436
\end{aligned}
\qquad (4.9)
$$

By simulating the above gains using Matlab®, a reasonable response can be achieved as shown in Figure (4.9).

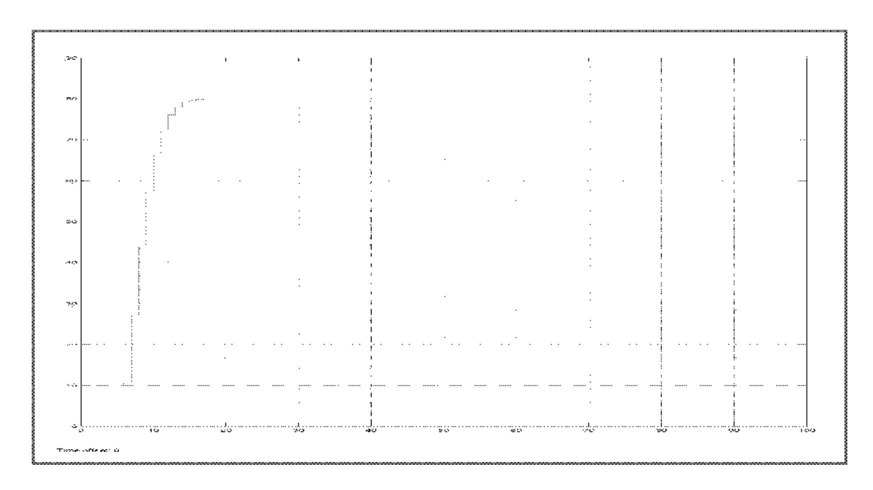


Figure 4.9 PIP-controller simulation using Matlab®

4.3 Practical Implementation

For the sake of implementation of the designed controller, a program using LabView® has been developed, see Figure (4.10). In this figure, the control law syntaxes using Math Box in LabView®. It is convenient to note that the sampling rate should be 4 seconds to match the sampling rate of collecting data.

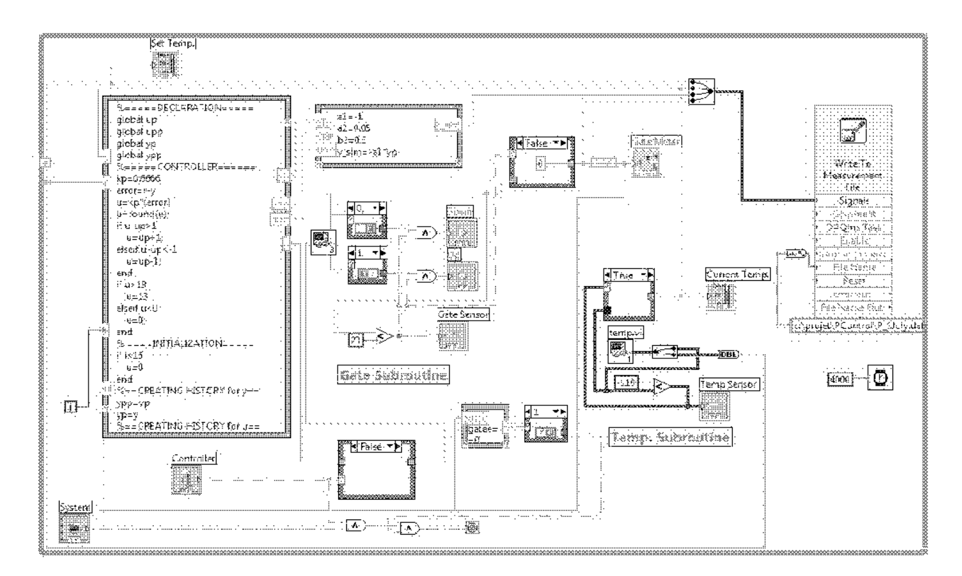


Figure 4.10 LabView® program used in implementation

The next figures shows the results and behaviour of the applied controllers for which Figure (4.11) shows the implementation of P-controller, Figure (4.12) shows the implementation of I-controller, Figure (4.13) shows the implementation of PI-controller, and finally Figure (4.14) shows the implementation of PIP controller.

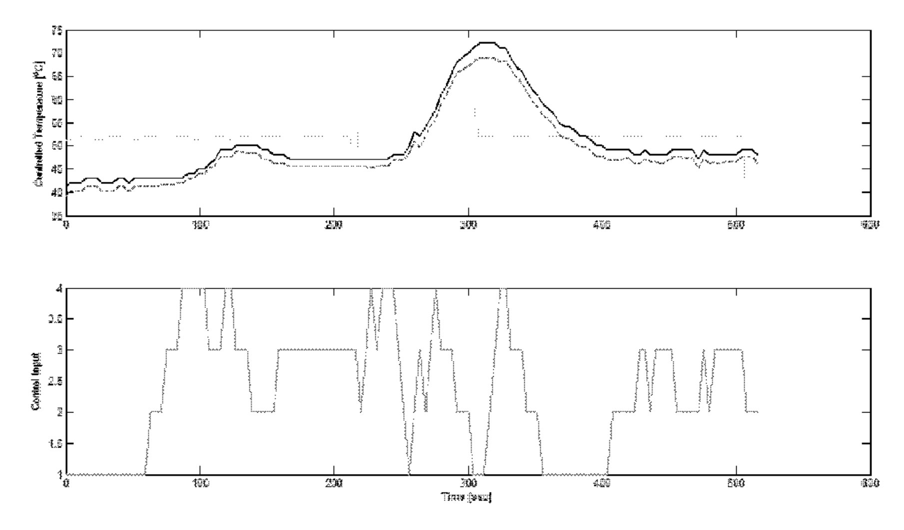


Figure 4.11 Practical response using P-controller

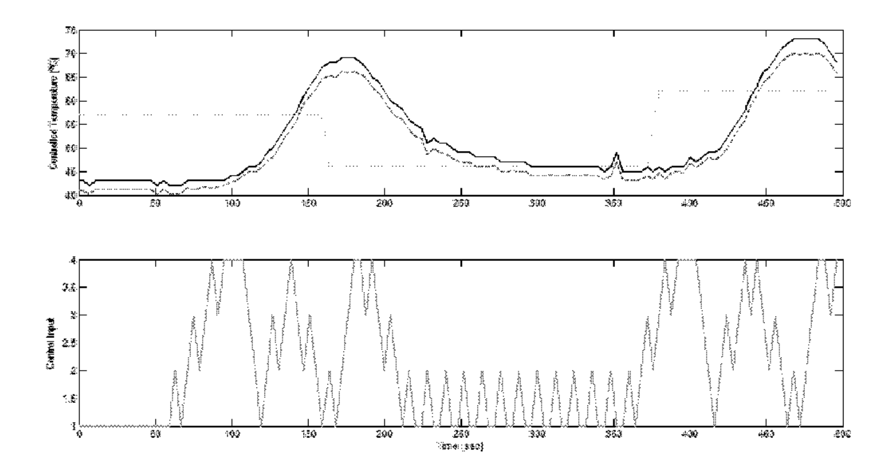


Figure 4.12 Practical response using I-controller

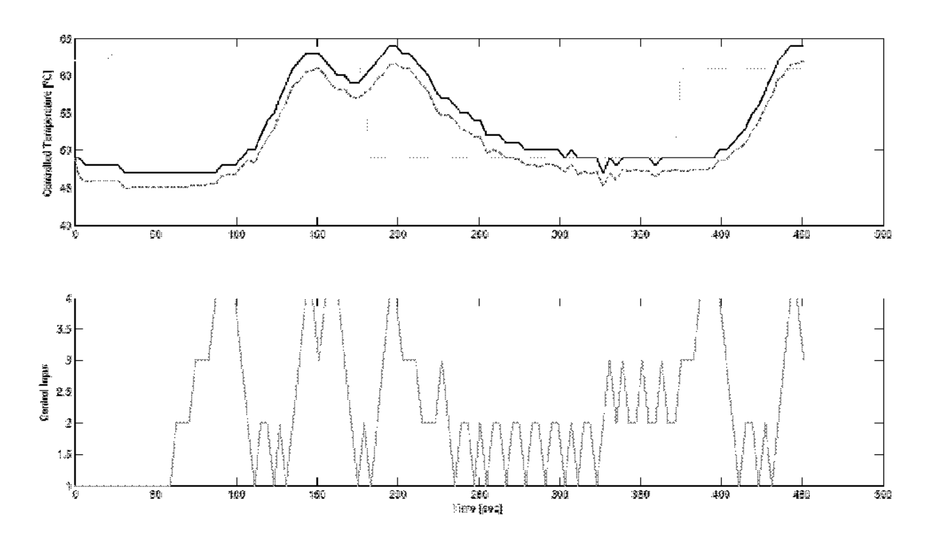


Figure 4.13 Practical response using PI-controller

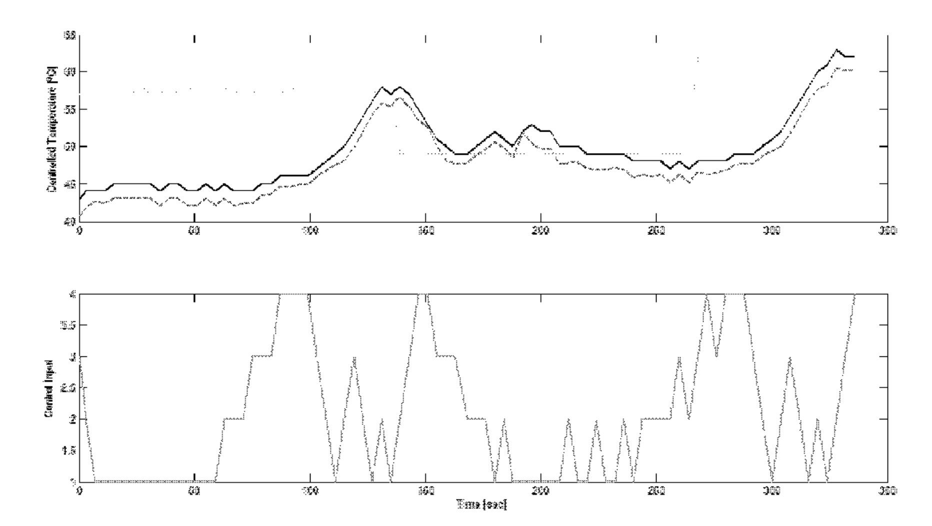


Figure 4.14 Practical response using PIP-controller

It is obvious that PIP gives the best performance among the applied controller. Also, PIP controller satisfies all the conditions stated in the problem statement.

Table of Contents

List of Figures

CHAPTER 5

TRACKING SYSTEM CONTROL

The second practical example is the automation tracking mechanism .Tracking is the process when we want to follow up the objective trajectory.

A DC motor has been used as an actuator and a simple NICKEL-Chrome wire has been used as potentiometer to feedback the current position of the tracker or cam. And another potentiometer is used to feedback the set point of the moving plate.

The target control system is automatically control the position of the moving plate (cam.) according to set point

Here, the system is SISO (single-input, single-output). The input is the position of the moving plate (set point), and the output is the position of the tracker or cam, see Figure (2.19). This work develops a complete control system and evaluates several algorithms for controlling the position of tracker or cam. It is convenient to note that the

implementation of PIP control (introduced in Chapter 3) represents one of the first application of such methods to a simple prototype in Egypt.

5.1 Dynamics for tracking System

Since the system is tracking, then it has relatively fast response for which it was found that 0.1sec sample delay is good enough to evaluate the dynamic behaviour of the system.

5.1.1 Linear model identification

In the first instance, linear TF models are identified and estimated using response error method by means *fminsearch* function existing in Matlab®. The coefficient of determination, R_T^2 is used to evaluate the estimated model. Step input functions are used to excite the system, with data collected for different magnitudes. The magnitudes are: u = 300, 600, 900, -300, -600 and -900. Also, one file was collected using random signal which ranges from $u = -1000 \rightarrow 1000$. In order to collect these data, a program using LabView® has been created as shown in Figure (5.1).

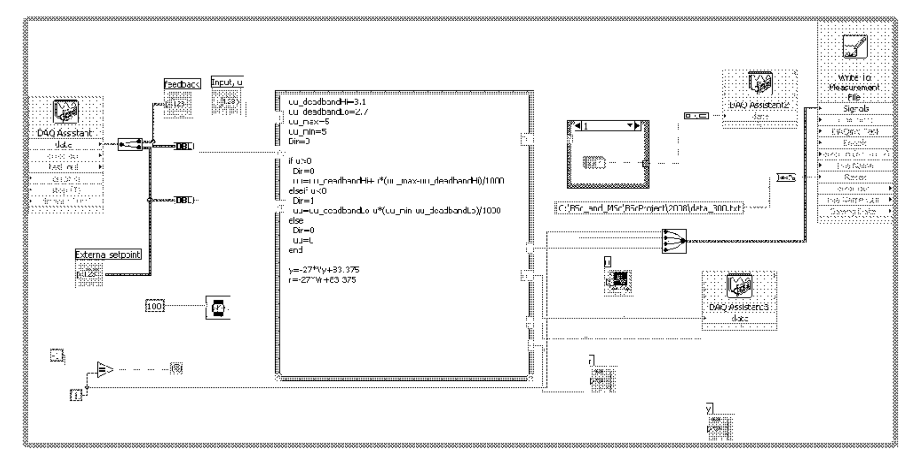


Figure 5.1 LabView® program used for collecting data

There are several models which can be well-define the dynamic behaviour of the system. In order to select the model which best define the system without over-parameterisation, the coefficient of determination, R_T^2, is utilised. It is found that model (1-1-2) is good enough to describe the dynamic behaviour of the system. Therefore, the model will take the form,

$$y_k = \frac{b_2 z^{-2}}{1 + a_1 z^{-1}} u_k \tag{5.1}$$

5.1.2 Linear model estimation

The next step is estimating the parameters of the estimated model (5.1). By means of the collected data several model parameters are achieved according to u. The values of these parameters are tabulated below.

u	300	600	900	-300	-600	-900
a_1	-0.9973	-0.9994	-1.0024	-1.0029	-1.0018	-1.0026
b_2	0.0011	0.0013	0.0013	0.0008936	0.0013	0.0014
R_T^2	0.9724	0.9694	0.9377	0.8711	0.9636	0.9651

The above table shows that the average values for a_1 is -1 and the average value for b_2 is 0.0012. Therefore, the model that best define the dynamic behaviour of the system is

$$y_k = \frac{0.0012 z^{-2}}{1 - z^{-1}} u_k \tag{5.2}$$

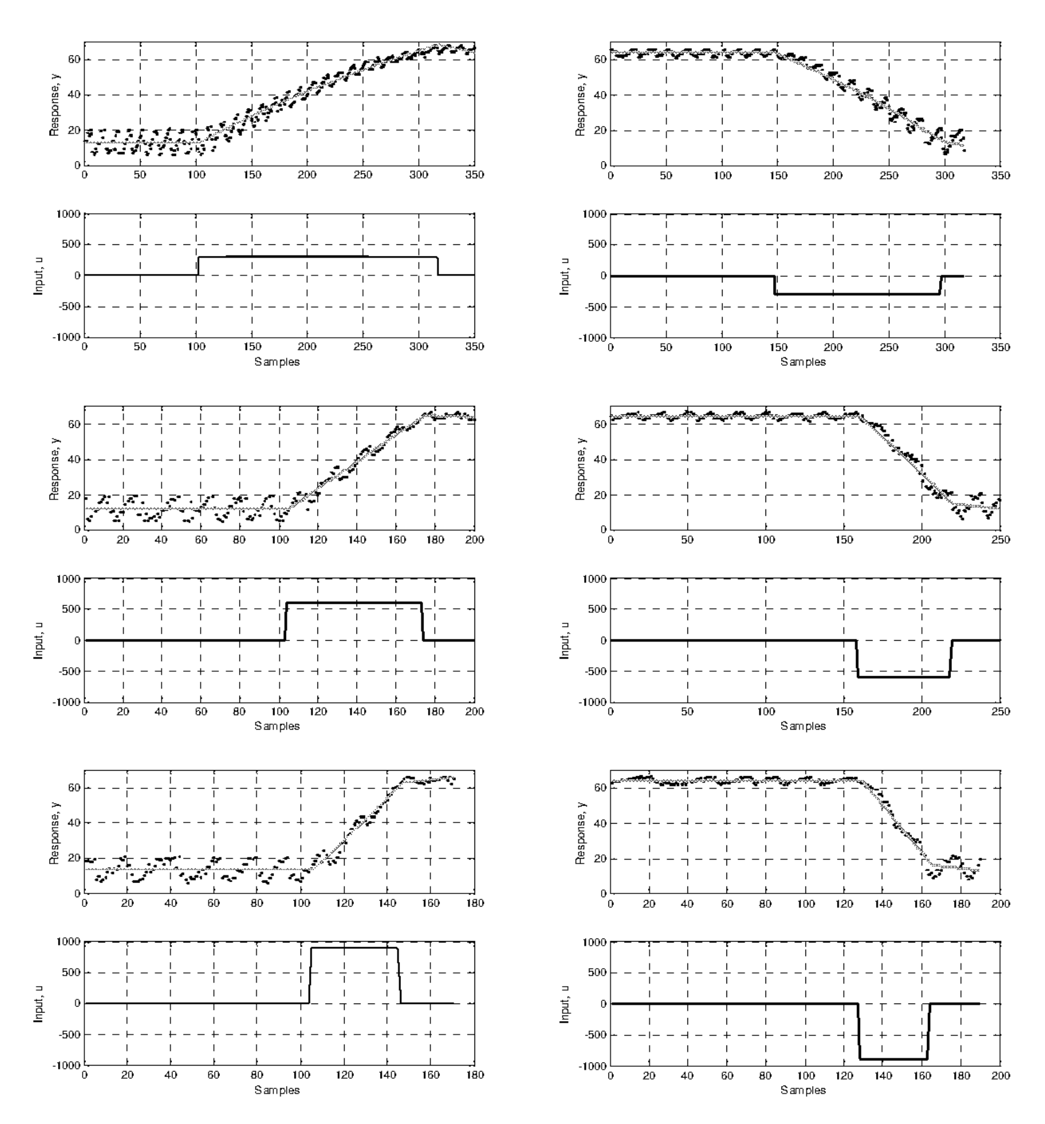


Figure 5.2 Model response for each data collected

5.2 P-, I-, PI- and PIP Controller Design

In this subsection, P-, I-, PI- and PIP controllers are developed for tracking system. Here, the control gains are determined by using pole assignment method.

5.2.1 P-Controller Design

Here, the control law will take the form of equation (3.18). The proportional gain k_p can be selected utilising the closed-loop TF depicted in Figure (3.4). By selecting poles $p_1 = p_2 = 0.5$, it is found that $k_p = 208.333$. This gain gives reasonable simulation results as shown in Figure (5.3) below.

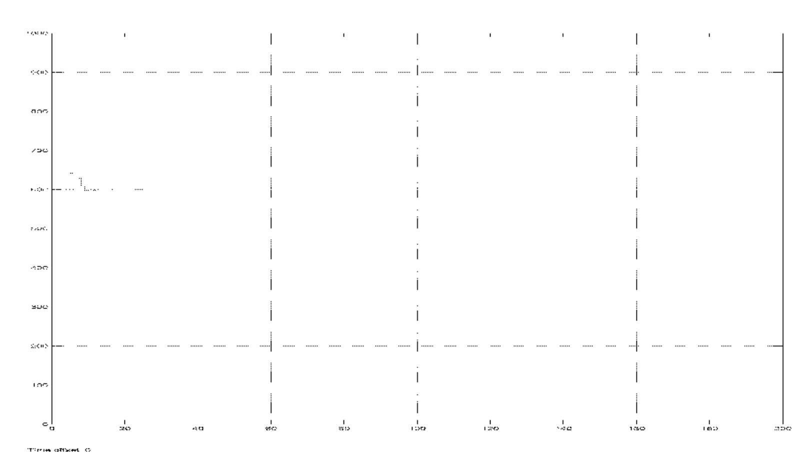


Figure 5.3 Simulation response using P-controller

5.2.2 I-Controller Design

It is impossible to design an I-controller for this system because its model is working as an integrator. In other words, $a_1 = -1$. The design of I-controller will lead to gains which lie outside the unit circle.

5.2.3 PI-Controller Design

Here, the control law will take the form of equation (3.21). The proportional and integral gains k_p and k_I can be selected utilising the closed-loop TF depicted in Figure (3.17). By selecting poles $p_1 = p_2 = 0.9$ and $p_3 = 0.2$, it is found that $k_p = 135$ and $k_I = 6.6667$. These gains give reasonable simulation results, see Figure (5.4).

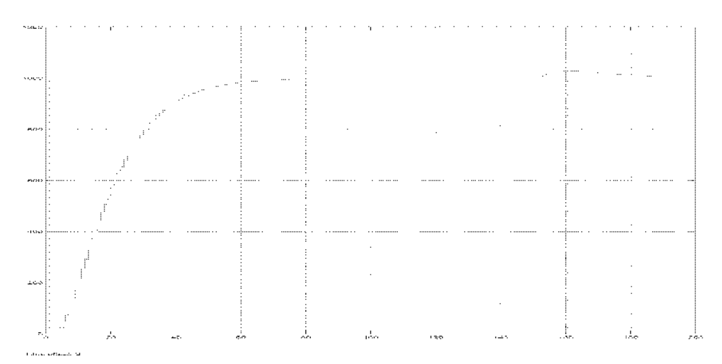


Figure 5.4 Simulation response using PI-controller

5.2.4 PIP Controller Design

Here, the SVF control law will take the form of equations (3.36 and 3.37). The proportional gains k_{p_1} and k_{p_2}, input gain k_u, and integral gain k_I can be selected utilising the closed-loop TF depicted in Figure (3.39). Here, the characteristic equation will be

$$\begin{aligned} D(z^{-1}) = 1-(2-k_u)z^{-1} - (2k_u - 0.0012k_p - 0.0012k_I - 1)z^{-2} \\ + (k_u - 0.0012k_p)z^{-3} \end{aligned} \tag{5.3}$$

which can be rewritten as

$$\begin{aligned} D(z) = z^3 - (2-k_u)z^2 - (2k_u - 0.0012k_p - 0.0012k_I - 1)z \\ + (k_u - 0.0012k_p) \end{aligned} \tag{5.4}$$

The poles should have the following form

$$(z-p_1)(z-p_2)(z-p_3) = z^3 - (p_1+p_2+p_3)z^3 + (p_1p_2 + p_2p_3 + p_1p_3)z^2 - p_1p_2p_3 \tag{5.5}$$

By comparing equation (5.4) with equation (5.5), then

$$\begin{aligned} &p_1 + p_2 + p_3 = 2 - k_u \\ &p_1 p_2 + p_1 p_3 + p_2 p_3 = -(2k_u - 0.0012k_p - 0.0012k_I - 1 \\ &p_1 p_2 p_3 = -(k_u - 0.0012k_p) \end{aligned} \tag{5.6}$$

By solving system of equation depicted in equation (5.6) simultaneously, the gains corresponding to the selected poles can be calculated. The next table shows some gain system corresponding to some proposed poles. The gains corresponding to poles $p_1 = p_2 = p_3 = 0.7$ can be found as

$$\begin{aligned} &k_p = 93.333 \\ &k_u = -0.4 \\ &k_I = 6.6667 \end{aligned} \tag{5.7}$$

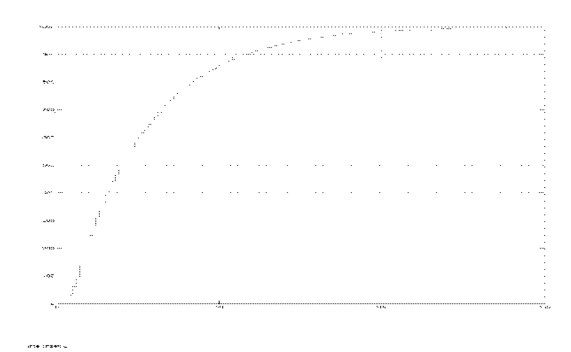


Figure 5.5 Simulation response using PI-controller

5.3 Practical Implementation

5.3.1 With refined feedback signal

For the sake of implementation of the designed controller, a program using LabView® has been developed, see Figure (5.6). In this figure, the control law syntaxes using Math Box in LabView®. It is convenient to note that the sampling rate should be 0.1 second to match the sampling rate of collecting data.

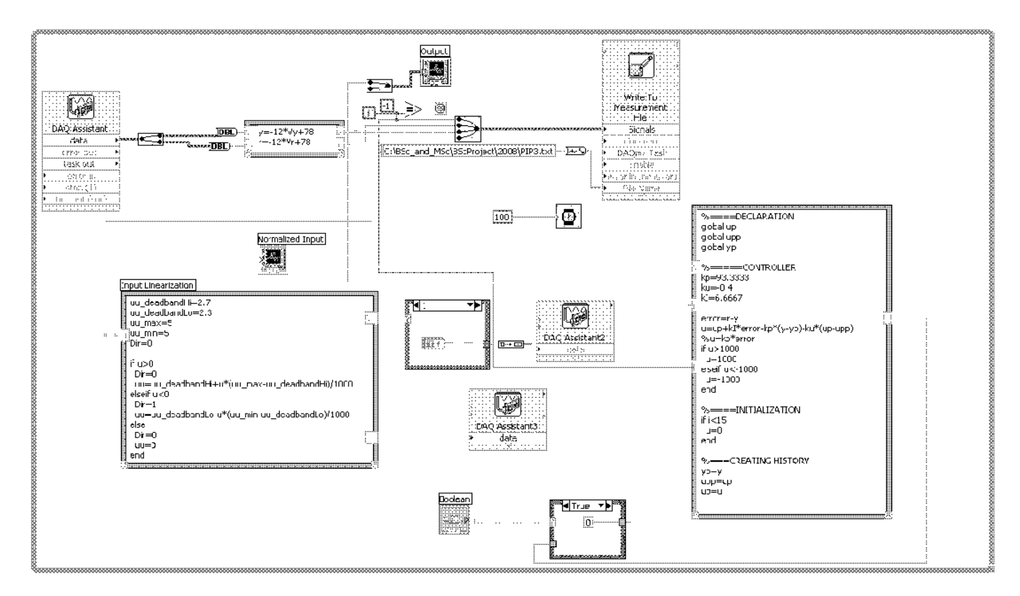


Figure 5.6 LabView® program used in implementation

The next figures shows the results and behaviour of the applied controllers for which Figures (5.7, 5.8, and 5.9) show the implementation of P-controller, PI-controller, and PIP controller, respectively.

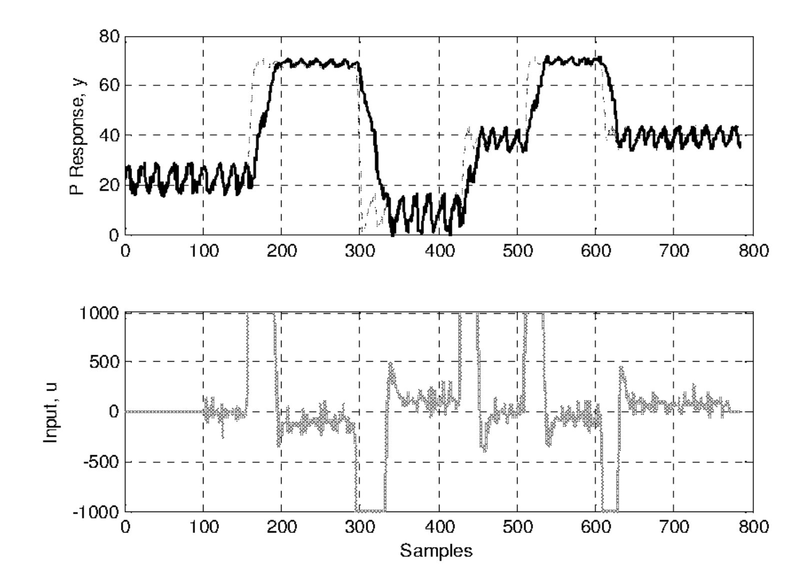


Figure 5.7 Practical response using P-controller

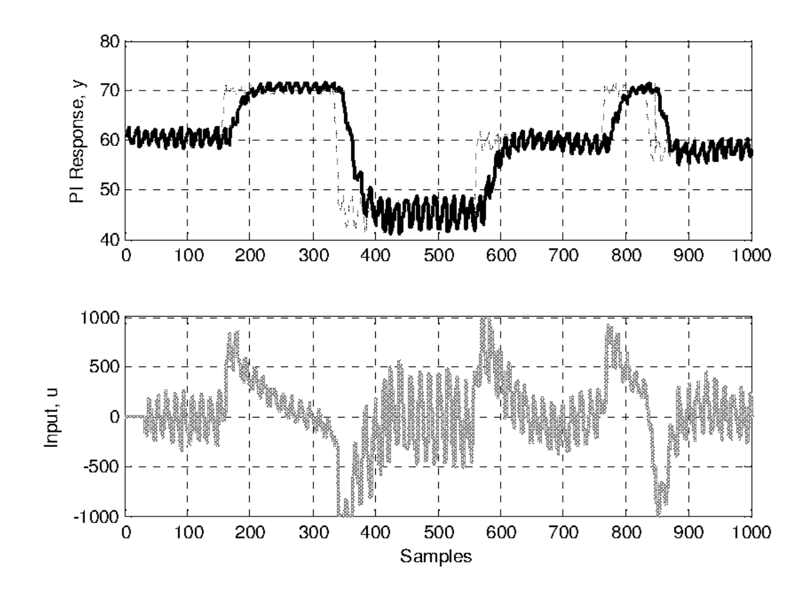


Figure 5.8 Practical response using PI-controller

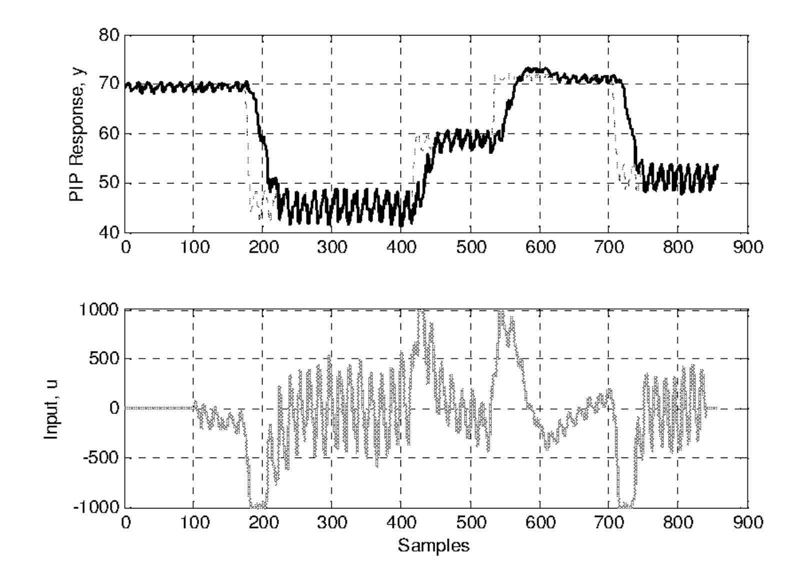


Figure 5.9 Practical response using PIP-controller

It is obvious that PIP gives the best performance among the applied controller. Also, PIP controller satisfies all the conditions stated in the problem statement.

5.3.2 With original feedback signal

The results achieved in the previous section have been carried out after refining the feedback signal[1]. The next figures show the response of the system before refining the signals coming from the potentiometer. As shown, the PI- and PIP control show very bad behaviour since their control law rely on the feedback signal of the current instant and its histories. Since, the noise in the current instant definitely is not the same as the noise of its history, the system is continuously suffers from fake output disturbances.

However, P- controller shows good behaviour since its control law depends upon the error between both the set point and the current feedback signal. It should be noted here, that the noise of both signals are the same since they have the same source of power. It should be noted here that in case of real output disturbance, the P controller will suffer from steady state error since one of the major drawback of P-controller that it cannot overcome input/output disturbances.

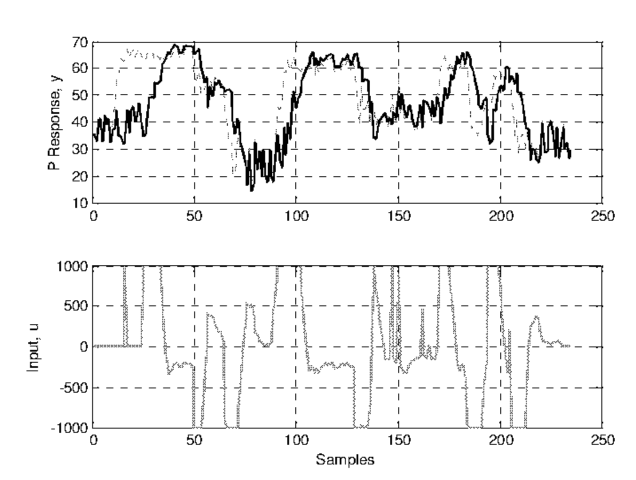


Figure 5.10 Practical response using P-controller using noisy feedback signals

[1] Refining feedback signals have been carried out by means of voltage divider which reduces the supplied voltade to the potentiometer from 5.7 volts to 2.8 volts.

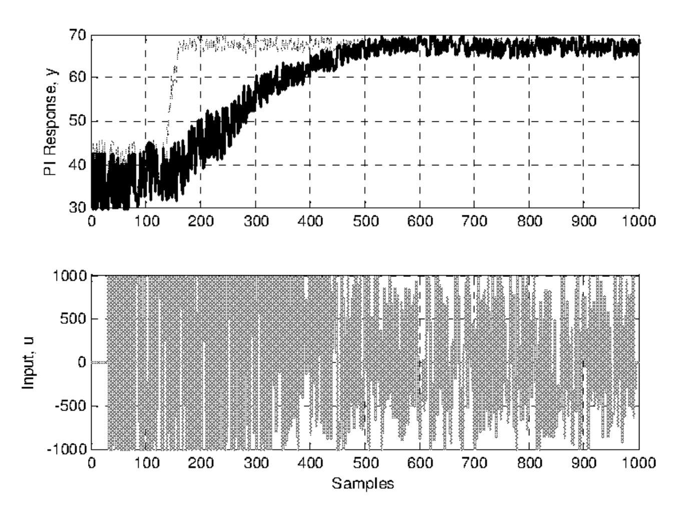


Figure 5.11 Practical response using PI-controller using noisy feedback signals

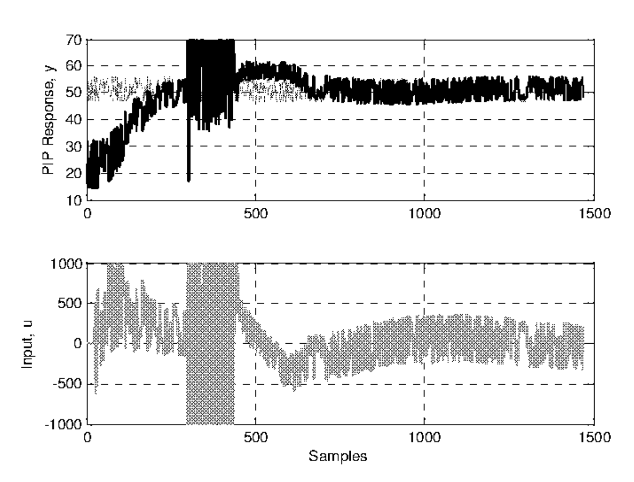


Figure 5.12 Practical response using PIP-controller using noisy feedback signals

Figure (5.10) shows the behaviour of P-controller in case of noisy feedback signals. Since these noise is not feedback, it has no effect on the actuator signal. However, in case of PI controller, these signals are feedback, therefore, it has a vibratory effect on the

actuator, see Figure (5.11). It should be noted here that both controllers cannot overcome output disturbances efficiently, but PI can overcome output disturbances partially.

As shown in Figure (5.12), however, PIP shows very oscillatory behaviour, it still has the capability to overcome output disturbance. But the oscillation in the actuator still has an effect of vibratory behaviour around the set point.

Table of Contents

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

A complete study is carried out during the creation of this work; starting with the new methodology of control which called PIP discussed in Chapter 3 and ending with its successful implementation for the first time in Egypt in real foundation (Cellopack Company located in 6th of October City) as discussed in Chapter 4. Also, an idea of tracking a moving object also discussed in Chapter 5. Finally, this chapter focuses the conclusion of the PIP approach used, results, and then ends with some recommendations for future work which based on the simulation and practical examples.

6.1 PIP Approach

The main aim of the thesis is to investigate PIP approach in control design of any dynamic system. In this thesis, two different demonstrators are used. The first is based on heat transfer system to control the temperature of hot air. The second is based on mechanical system to control the position of the tracker to follow a certain tracked object. Therefore, the development of PIP algorithm based upon the NMSS formulation is introduced along with illustrative examples which are employed to emphasize the

difference between the PIP and the conventional P-, I-, and PI control algorithms (Chapter 3). Pole placement is explained and utilised to find the SVF gain vector.

The work also considers the limitations of the conventional approaches (P-, I- and PI) through illustrative examples. These limitations are arisen due to several reasons such as higher order of the TF, higher sample delay and/or lack of gains. For this reason, PIP is considered to overcome these difficulties to attain the required set point. Over the last few years, such PIP control systems have been successfully employed in a wide range of applications, including construction: Seward *et al.* (1997); Gu *et al.* (2004). The preliminary simulation results of illustrated examples suggest that PIP approach offers considerable improvement over P-, I-, PI- designs, so providing smoother, more accurate tracking to the set points, and faster response to the dynamic system. Monte Carlo analysis suggests that PIP design is more robust to model uncertainty, as illustrated in simulation of the demonstrators.

The above results have been confirmed through practical implementation of the approach using the two different practical examples. However, sometimes PIP loose controllability when the feedback signals are noisy. In this case, primitive controllers was found more accurate and give satisfactory closed-loop performance. Therefore, improving signal quality and reduce the noise is essential in order to get good performance for PIP controller, see Chapter 5.

As shown in the demonstrator examples, the PIP control algorithm shows better performance. For example, in the hot air system, the set point can be reached by means of PIP controller in lea than 1 min (the required is 30 minumtes), and the system shows very good behaviour in both input and output disturbances.

In case of mechanical tracking system; the P-controller shows better performance over PIP, when the feedback noise is very large. However, by reducing the noise, PIP shows better performance over the other controllers, as expected.

6.2 Implementations and Achievements

The implementations chapters are included to investigate the earlier methodological approach as well as provide the industrial sector with initial ideas for controlling the temperature and tracking a moving object. In this regard, primarily, two main practical examples related to heat transfer and mechanical system used for this purposes are introduced, since automation tracking and observing objects are now starting to be adopted as a means of improving efficiency and quality. The first example is the real (full scale) Hot Air Temperature Control (Chapter 4), the second is the Position Control for tracking an object which is ideal for the development and evaluation of the control method (Chapter 5). Finally, hardware and software developments of these two examples are introduced in order to achieve a realistic prototype for industrial sectors.

The proposed applications provide fertile prototype to illustrate the feasibility and applicability of the P-, I-, PI-, and PIP controller designs, as well as comparing their performance characteristics. It should be noted here that both examples are based on SISO control algorithms.

6.2.1 Hot air temperature control

Chapter 4 describes the control of a hot air temperature developed for Cellpack Company (6^{th} of October branch). In this regard, the present work considers the implementation of the complete control system for automatically controlling the temperature of the hot air to dry liquid-based ink for a huge printing machine. All

methodological approaches discussed in the thesis are evaluated, e.g. P-, I-, PI- and PIP control.

In the first instant, PIP approach gives improved closed-loop performance. In particular, the time taken to reach the set point is considerably reduced. In general, PIP control yields smoother, more robust closed-loop characteristic. However, by selecting proper gain values, the other controllers (P-, I-, and PI) can give similar response but in bigger settling time.

Also, the selection of digital encoder is found impractical, since the gate opening cannot achieve a decimal value. This restriction causes the system to oscillate about the set point.

6.2.2 Tracking system control

Two phases of implementation are carried out through this practical example, as illustrated in Chapter 5. In the first phase, a comparison between control methodologies whereas the system feedback very high noisy signals. However, in the second, an evaluation of the same methodologies is considered after refining the feedback signals.

Regarding the first phase (high noisy feedback signals), P-control yields better response since there is no feedback from the noisy signals is utilised. In this regard, other control methodologies (PI- and PIP) give very bad response. However, I-controllers cannot be applied at all since the system is working as an integrator. It should be noted here that still P-controller cannot overcome output disturbances if happened.

The second phase, where the feedback signals are refined, shows successful implementation of PIP and other controllers. However, PIP algorithm shows better performance among the other methodologies. The results show that the control system increases the speed of operation and the system can now overcome any input/output disturbances.

6.3 Recommendations and Future Work

Recommendations for future work can be divided into two main categories, they are:

- **Recommendations for methodological approaches:** In this regard, the following recommendations are provided for future work:

  1- PIP approach has the ability to get feedback from the input signal. Therefore, a sensor that measure these signals should be installed for this purpose. This is happened in case of hot temperature control by means of the encoder mounted on the shaft of the DC motor of the natural gas gate. However, in case of the tracking mechanism, the input feedback is taken from the program rather than the hardware itself.

  2- In order to apply PIP or any other methodology that based on feedback the signal and its history require good enough or high quality signal with acceptable noise. Otherwise, bad behaviour will appear.

- **Recommendations for practical examples:** The following recommendations are provided for future work based on the implementation processes of the practical examples:

1- It is preferable to replace the digital encoder with potentiometer, so a decimal value of the gate opening can be read rather than an integer value. This would help to open the gate a fraction of the unity if required.

2- It is preferable to control the amount of the fresh air introduced into the system as well, this will help to economise the cost of the natural gas consumption. This will lead to multivariable control system which is beyond the topic of the project.

3- It is preferable to increase the degree of freedom of the tracking system to be two-degrees of freedom. This will increase the flexibility of tracking. image processing software should be embedded in order to detect the object and then replace the simple plate.

TABLE OF CONTENTS

LIST OF FIGURES

RESPONSE ERROR
MATLAB® Files

M-File

```
close all
clear all
clc

S=load('c:\data\CollectedData\data_01.dat');

y=S(:,2);
m=y(1);
y=y-m;
u=S(:,3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%
% Optimising Parameters Using "fminsearch" function %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%
alpha0(1)=0;% alpha0(2)=0;

alphaf=fminsearch('TF_function',alpha0,[],y,u)

[lse,y_est]=TF_function(alphaf,y,u);
y=y+m;
y_est=y_est+m;
Rt2=1-cov(y_est-y)/cov(y)
%PLOTTING
figure(1)
subplot(211)
plot(y,'.k');hold on;plot(y_est,'-b','linewidth',2)
ylabel('Response')
```

```
grid on

subplot(212)
plot(u,'-k','linewidth',2)
ylabel('Input')
xlabel('Samples')
grid on
%legend('Data points','Model estimated using SRIV')
```

Function

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Fuction used for "fminsearch" optimization %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [lse,y_est]=TF_function(alpha,y,u)
%*****************************************************************
%* lse = the sum of Least Squares Errors *
%* (objective function criteria for FMINSEARCH) *
%* rt2 = coefficient of determination *
%* (based on simulation response) *
%* y_est = Simulation Response (Transfer Function Response) *
%*****************************************************************

%*****************************************************************
%* The function used now for evaluating is: *
%* y(k)=-a1*y(k-1)-a2*y(k-2)-a3*y(k-3)+b2*u(k-2) *
%* where; *
%* a1 = alpha(1) a2 = alpha(2) a3 = alpha(3) *
%* b2 = alpha(4) *
%*****************************************************************
y_est(1:2)=y(1:2);
for k=3:length(u)
   y_est(k)=0.95*y_est(k-1)+alpha(1)*u(k-2);
end

y_est=y_est';
lse=y-y_est;;
lse=lse'*lse;
```

REFERENCES

1- Adapted from Modern Control Theory System, 9th ed., R.C. Dorf and R.H. Bishop, Prentice-Hall, 2001.

2- Dixon, R. and Lees, M.J., (1994), Implementation of Multivariable Proportional-Integral-Plus (PIP) Control Design for a Coupled Drive System, *Proceedings of 10th International Conference on Systems Engineering (ICSE)*, Vol.1, pp.286-293, Coventry University, UK.

3- Dixon, R., Chotai, A., Young, P.C., Scott, J.N., (1997), The Automation of Piling Rig Positioning Utilising Multivariable Proportional-Integral-Plus (PIP) Control. *In: Proceedings 12th International Conference on Systems Engineering (ICSE)*, 9-11 September, Coventry University, UK.

4- Dixon, R., Taylor, C.J. and Shaban, E.M., (2005), Comparison of classical and modern control applied to an excavator-arm, *International Federation of Automatic Control 16th Triennial World Congress (IFAC06),* 4-8 July, Prague, Czech Republic.

5- Gu, J., Taylor, C.J. and Seward, D., (2004), Proportional-Integral-Plus Control of an Intelligent Excavator", *Journal of Computer-Aided Civil and Infrastructure Engineering*, Vol.19, pp.16-27.

6- Seward, D. and Quayle, S., (1997), LUCIE - Lancaster University Computerised Intelligent Excavator. *Circ. Cellar: Comput. Appl. J.*, Vol.79, pp.67-73.

7- Shaban, E.M., Taylor, C.J. and Ako, A., (2006), Development of an automated verticality alignment system for a vibro-lance, *In preparation for submission to IEE Proceedings.*

8- Taylor, C.J., McCabe, A.P., Young P.C. and Chotai, A., (2000b), Proportional-Integral-Plus (PIP) control of the ALSTOM gasifier problem, *Proceedings of the Institution of Mechanical Engineers, Part I, Journal of Systems and Control Engineering*, Vol.214, No.16, pp.469-480.

9- Taylor, C.J., Young, P.C., Chotai, A., Mcleod, A.R. and Glasock, A.R., (2000c), Modelling and Proportional-Integral-Plus Control Design for Free Air Carbon Dioxide Enrichment Systems, *Journal of Agricultural Engineering Research,* Vol.75, pp.365-374.

10- Taylor, C.J., Leigh, P., Price, L., Young, P.C., Berckmans, D., Janssens, K., Vranken, E. and Gevers, R., (2004a), Proportional-Integral-Plus (PIP) control of Ventilation rate in agricultural buildings, *Control Engineering Practice*, Vol.12, No.2, pp.225-233.

11- Taylor, C.J., Leigh, P.A., Chotai, A., Young, P.C., Vranken, E. and Berckmans, D., (2004b), Cost Effective Combined Axial Fan and Throttling Valve Control of Ventilation Rate, *Proceedings of the Institution of Electrical Engineers, Control Theory and Applications,* Vol.151, No.5, pp.577-584.

12- Young, P.C, Lees, M., Chotai, A., Tych, W. and Chalabi, Z.S., (1994), Modelling and PIP Control of a Glasshouse Micro-Climate, *Control Engineering Practice*, Vol.2, pp.591-604.